RISE3301 数据手册
JS-000317
Version 1.1

瑞斯康微电子(深圳)有限公司

目　录

第一章　RISE3301 概述……6

1.1　特点与优点……6

1.2　硬件资源和参数指标……6

第二章　引脚说明……3

2.1 引脚的分类……3

2.2 引脚功能描述……4

2.3 封装参数……10

第三章　存储器配置……11

3.1 程序存储器配置……11

3.2 数据存储器配置……12

3.3 片内数据存储器配置……15

3.4 片内 SRAM 的分配……17

3.5 Mail Box 功能说明……18

第四章　电力载波发送和接收模块……19

4.1 描述……19

4.2 RISE3301 载波发送模块的原理及特性……19

4.3 RISE3301 载波接收模块的原理及特性……20

第五章　系统时钟……21

5.1 描述……21

5.2 系统时钟介绍……21

第六章　中　断……22

第七章　GPIO 口……23

7.1 GPIO 口描述……23

7.2 复用口介绍……23

7.3 GPIO 口寄存器……24

第八章　通讯口/Timer 功能……26

8.1 UART 口……26

8.2 I2C 通讯口……26

8.3 SPI 通讯口……*32*

8.4 Timer 定时器……*36*

第九章 内部 RTC……**37**

9.1 硬件原理……*37*

9.2 功能描述……*37*

9.3 工作寄存器……*38*

第十章 复位……**43**

10.1 外围硬件复位……*43*

10.2 底层软件控制复位……*43*

10.3 WDT 复位……*44*

10.4 复位寄存器……*44*

第十一章 电气特性……**46**

11.1 工作参数……*46*

11.2 极限参数……*46*

11.3 DC 特性……*46*

11.4 时序特性……*47*

附录……**52**

图目录

图 1-1 内部功能模块图 7
图 2-1 引脚定义图 3
图 2-2 封装参数图 10
图 3-1 程序存储器空间地址分配图 11
图 3-2 数据存储器空间地址分配图 12
图 3-3 MEMORY 读写时序图 14
图 3-4 片内 256 个 RAM 单元分配 15
图 3-5 片内 SRAM 分配图 17
图 3-6 MAIL BOX 功能图 18
图 4-1 载波发送底层硬件原理图 19
图 4-2 载波接收底层硬件原理图 20
图 5-1 系统时钟图 21
图 8-1 I^2C 原理图 27
图 8-2 SPI 端口图 33
图 9-1 内部 RTC 原理图 37
图 10-1 外围硬件复位信号波形 43
图 10-2 复位图 43
图 11-1 I^2C 定时 48
图 11-2 SPI 定时 49

表目录

表格 2-1 引脚分类 3
表格 2-2 电源引脚 4
表格 2-3 接地引脚 4
表格 2-4 系统、时钟晶振模块引脚 4
表格 2-5 外扩存储空间地址总线 5
表格 2-6 外扩存储空间数据总线 5
表格 2-7 总线控制引脚 5
表格 2-8 外部中断、复位引脚 6
表格 2-9 SPI 口引脚 6
表格 2-10 UART 口引脚 6
表格 2-11 专用 GPIO 口引脚 7
表格 2-12 复用 GPIO 口引脚 7
表格 2-13 I^2C 口引脚 7
表格 2-14 载波接收模块引脚 7
表格 2-15 载波发送模块引脚 8
表格 2-16 工厂成品测试引脚 9
表格 2-17 DOCD 开发平台调试引脚 9
表格 2-18 测试引脚 9
表格 3-1 数据存储器对应分配表 12
表格 3-2 外部总线接口寄存器 13

表格 3-3 延时等待寄存器.....13
表格 3-4 EBIM 控制寄存器.....13
表格 3-5 MEMORY 接口读写时序表.....14
表格 3-6 特殊功能寄存器地址定义表.....15
表格 5-1 时钟信号表.....21
表格 7-1 复用口表.....23
表格 7-2 上标定义.....24
表格 7-3 GPIO 口寄存器.....24
表格 8-1 I^2C 模块寄存器综述（基本地址: 0X005000).....27
表格 8-2 控制寄存器位定义.....28
表格 8-3 状态寄存器.....29
表格 8-4 从设备地址寄存器.....30
表格 8-5 时钟分频寄存器功能.....30
表格 8-6 状态映射寄存器功能.....31
表格 8-7 数据缓冲区功能.....31
表格 8-8 TGSR 功能.....31
表格 8-9 SPI 寄存器.....34
表格 8-10 SPI 控制寄存器功能.....34
表格 8-11 波特率寄存器功能.....35
表格 8-12 SPI 收/发控制寄存器功能.....35
表格 8-13 状态寄存器功能.....35
表格 8-14 发送数据缓冲区功能.....36
表格 8-15 接收数据缓冲区功能.....36
表格 9-1 RTC 寄存器.....38
表格 9-2 计时寄存器功能.....39
表格 9-3 报警寄存器功能.....40
表格 9-4 BASE TIME 寄存器功能.....40
表格 9-5 RTC 控制寄存器功能.....40
表格 9-6 初始值寄存器功能.....41
表格 9-7 RTC 状态寄存器功能.....41
表格 9-8 停电计数寄存器功能.....41
表格 9-9 停电计数寄存器功能.....42
表格 10-1 复位寄存器位定义.....44
表格 10-2 系统 RESET 控制寄存器功能.....44
表格 11-1 工作参数表.....46
表格 11-2 极限参数表.....46
表格 11-3 DC 特性表.....46
表格 11-4 时钟定时.....47
表格 11-5 I^2C 接口定时.....48
表格 11-6 SPI 接口定时.....48

第一章 RISE3301 概述

RISE 3301 是一款可用于实现智能控制网络的单片系统芯片（SoC）。其主要特性是利用低压电力线作为通信媒介，实现数据传输及网络控制等功能。可广泛应用于远程抄表和负荷管理、工业自动化控制、安防系统、交通运输自动化以及家庭智能自动化等领域。

RISE3301 的设计符合 EIA_709.1，EIA-709.2 和 EN50065-1 等国际标准。同时增加了自动路由协议层。RISE3301 一其性能优越的物理层设计及完善的网络通信协议，保证了可靠的网络通信性能。它高集成化的 SoC 设计及通用而高效的开发平台能满足各个领域用户的应用开发需求。

1.1 特点与优点

RISE3301 芯片的主要特点与优点有：

❶ **性能优越的物理层设计**

RISE3301 芯片性能独特，具有脉冲干扰抑制功能，可实现快速信道自适应，快速自动增益控制。可将取样时间的误差自动纠正到 200PPM 以上。同时还具有监控接收信号质量的能力，可监控信号功率、干扰强度以及误码率等信息。

❶ **完善的网络通信协议**

RISE3301 芯片符合 EIA_709.1 和 EIA-709.2 标准规范的通信协议，可保证多节点网络高效及可靠的通信。RISE3301 芯片在设计时还增加了自动路由控制层，每个节点均可动态地被设置为路由器或转发器，使整个电力通信网络系统的性能更稳定。通过其自动建网、自适应及自动维护网络等特点，您不必再投入大量的人工干预以进行人工网络测量、调试等步骤，从而大大节省维护费用。

❶ **高集成化的 SoC 设计**

该芯片集成了双 CPU，整个通信协议栈功能由一个 CPU 完成，另一个 CPU 可被用来提供应用系统的编程；接收及发送数据处理（DSP）模块按照 EIA_709.1，EIA-709.2 和 EN50065-1 国际标准设计；配备有模数/数模转换器（ADC,DAC）及可编程增益放大器；此外，芯片还内置有 12KB 内存储器（SRAM），用于暂存数据包及应用程序变量。

❶ **通用而高效的开发平台、充足的用户应用开发资源**

RISE3301 芯片选用 Keil μVision 作为它的软件开发工具，将 KeilC 语言编辑、汇编编译、连接及调试等功能集于一体，用户可方便地使用 Keil μVision 来开发各种应用系统。内嵌便于用户应用开发的高速 CPU，实现各领域用户应用开发需求。

1.2 硬件资源和参数指标

RISE3301 的主要硬件资源和参数指标如下：

★ 内嵌两个高速 8 位 CPU，时钟频率达 80MHz，与 8051 系列 MCU 兼容；
★ 物理层通信速率：5.5Kbits/s
★ 可编程多频点载波通信

- 24 个软件可配置 GPIO
- 2 个软件可配置 UART
- Memory 接口可扩展 128 KB 外部 Flash Memory 和 256 KB 外部 SRAM；
- 6 个 16 位定时器、6 个外部中断输入；
- 载波信号调制方式：BPSK
- 12KB 内存储器（SRAM）
- 实时时钟（RTC）和看门狗
- 模数/数模转换器（ADC,DAC）及可编程增益放大器
- 多个外设通讯口（2 个 UART 口，1 个 SPI 口，1 个 I^2C 口等）
- 24 个软件可配置 GPIO 口；
- 内置 PLL 锁相环倍频电路；
- 采用 BPSK 调制解调技术；电力线数据通讯波特率可达 5.5kbps；
- COMM 微处理器中嵌入并固化有 EIA709.1 协议栈和系统自动路由算法模块

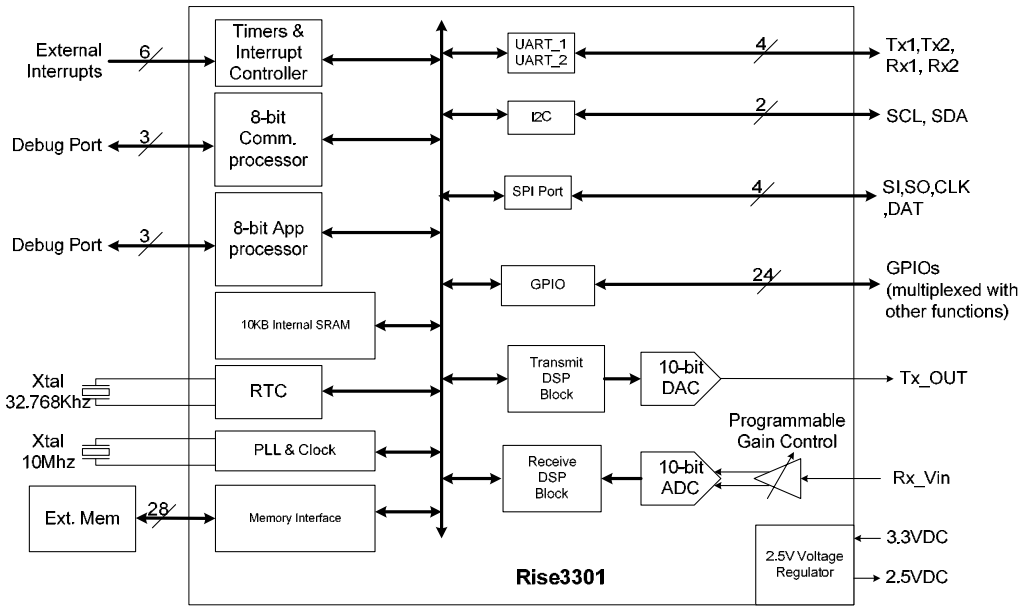


图 1-1 内部功能模块图

第二章 引脚说明

2.1 引脚的分类

RISE3301 芯片引脚如下图所示：

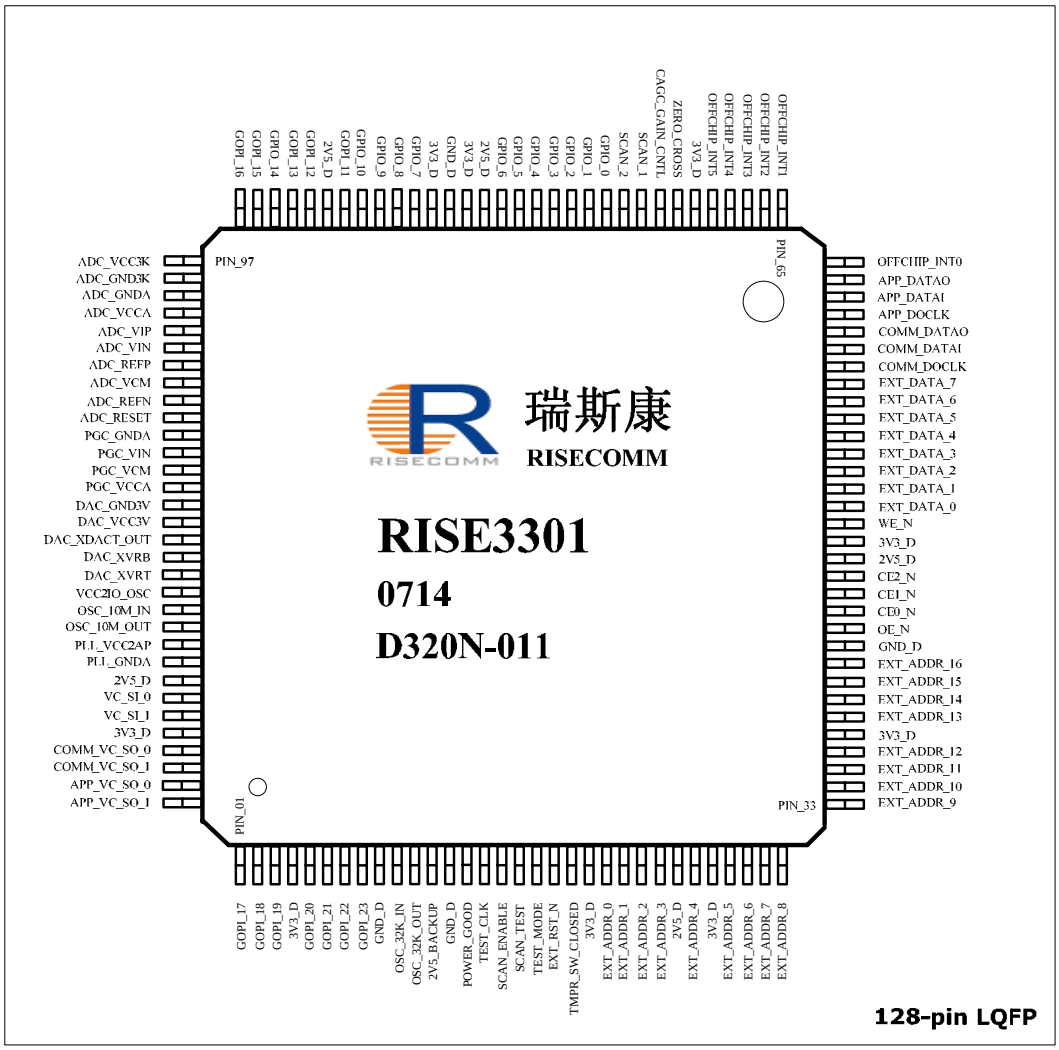


图 2-1 引脚定义图

RISE3301 芯片 128 引脚可根据功能特性分为以下多种类型（参见下表）。

表格 2-1 引脚分类

引脚类型	引脚总数	表格
电源、接地引脚	(9、6、1、4)	表格 2-2、2-3
系统、时钟晶振模块引脚	(4、4)	表格 2-4
控制引脚	（17、8、5、6、1、24）	表格 2-5、2-6、2-7、2-8、2-9、2-10、2-11、2-12、2-13

载波电路模块引脚（接收、发送）	（16、5）	表格 2-14、2-15
测试、调试引脚	（6、6、6）	表格 2-16、2-17、2-18

2.2 引脚功能描述

各类引脚的具体功能描述请参见下表。

2.2.1 电源、接地引脚

表格 2-2 电源引脚

引脚数	Pin	名称	类型	方向	功能
9	4、21 、28 、37、48、70、83、85、124	**Vio**	Power	In	3.3V 数字 I/O 口供电电源；
5	26、47、91、116、121	**Vck**	Power	In	2.5V 数字核供电电源；
1	82	**Vck**	Power	Out	2.5V 数字核供电电源输出（仅供本芯片用）；跨个 3.3uF 电容到地：
1	12	**Vbk**	Power	In	作为芯片内核工作电平同时也作为内部 RTC 电源 2.5VBK；

表格 2-3 接地引脚

引脚数	Pin	名称	类型	方向	功能
4	9 、13 、42、84	**GND**	Power	In	数字地

2.2.2 系统、时钟晶振模块管脚

表格 2-4 系统、时钟晶振模块引脚

引脚数	Pin	名称	类型	方向	功能
1	117	**OSC_10M_IN**	Dig	In	系统晶振输入；
1	118	**OSC_10M_OUT**	Dig	Out	系统晶振输出；
1	119	**PLL_VCC2AP**	Power	In	PLL 电源 2.5V；
1	120	**PLL_GNDA**	Power	In	PLL 模拟地；
1	10	**OSC_32K_IN**	Dig	In	RTC 晶振输入；
1	11	**OSC_32K_OUT**	Dig	Out	RTC 晶振输出；
1	14	**POWER_GOOD**	Dig	In	停电检测引脚； 当主电源掉电，该引脚产生电平变化（高-低）则启动停电 RTC 记时功能。RTC 记录停电时间；
1	20	**TMPR_SW_CLOSED**	Dig.BU	In	记录外部事件； Tamper switch 模块检测到该引脚电平变化（高-低）则启动 Tamper switch Counter 记时功能，计时周期为 RTC 计时时基；

2.2.3 控制引脚

表格 2-5 外扩存储空间地址总线

引脚数	Pin	名称	类型	方向	功能
17	22	**EXT_ADDR_0**	Dig	Out	17 根外扩存储空间地址线；
	23	**EXT_ADDR_1**			
	24	**EXT_ADDR_2**			
	25	**EXT_ADDR_3**			
	27	**EXT_ADDR_4**			
	29	**EXT_ADDR_5**			
	30	**EXT_ADDR_6**			
	31	**EXT_ADDR_7**			
	32	**EXT_ADDR_8**			
	33	**EXT_ADDR_9**			
	34	**EXT_ADDR_10**			
	35	**EXT_ADDR_11**			
	36	**EXT_ADDR_12**			
	38	**EXT_ADDR_13**			
	39	**EXT_ADDR_14**			
	40	**EXT_ADDR_15**			
	41	**EXT_ADDR_16**			

表格 2-6 外扩存储空间数据总线

引脚数	Pin	名称	类型	方向	功能
8	50	**EXT_DATA_IO_0**	Dig	BiDi-PU	并行 8 根数据线；
	51	**EXT_DATA_IO_1**			
	52	**EXT_DATA_IO_2**			
	53	**EXT_DATA_IO_3**			
	54	**EXT_DATA_IO_4**			
	55	**EXT_DATA_IO_5**			
	56	**EXT_DATA_IO_6**			
	57	**EXT_DATA_IO_7**			

表 格 2-7 总线控制引脚

引脚数	Pin	名称	类型	方向	功能
1	43	**OE_N**	Dig	Out	外围存储器开使能，低电平有效；
1	44	**CE0_N**	Dig	Out	外围存储器片选信号； 低电平有效；用于选择连接 FLASH ；
1	45	**CE1_N**	Dig	Out	外围存储器片选信号； 低电平有效；用于选择连接 SRAM；
1	46	**CE2_N**	Dig	Out	外围存储器片选信号； 低电平有效；用于选择连接 SRAM；

1	49	**WE_N**	Dig	Out	外围存储器写使能，低电平有效；

表 格 2-8 外部中断、复位引脚

引脚数	Pin	名称	类型	方向	功能
6	64	**OFFCHIP_INT_0**	Dig	In	外部中断；下降沿触发，低电平有效时间不得低于 25ns；
	65	**OFFCHIP_INT_1**			
	66	**OFFCHIP_INT_2**			
	67	**OFFCHIP_INT_3**			
	68	**OFFCHIP_INT_4**			
	69	**OFFCHIP_INT_5**			
1	19	**EXT_RST_N**	Dig	In	硬件复位；低电平有效；

表格 2-9 SPI 口引脚

引脚数	Pin	名称	类型	方向	功能
1	79	**SCLK**	Dig	BiDi	SPI 时钟信号线；
		GPIO_4			作为普通 I/O 使用；
1	80	**MOSI**	Dig	BiDi	SPI 数据口；
		GPIO_5			作为普通 I/O 使用；
1	81	**MISO**	Dig	BiDi	SPI 数据口；
		GPIO_6			作为普通 I/O 使用；
1	86	**SS_0**	Dig	BiDi	从设备片选；
		GPIO_7			作为普通 I/O 使用；
1	87	**SS_1**	Dig	BiDi	从设备片选；
		GPIO_8			作为普通 I/O 使用；
1	88	**SS_2**	Dig	BiDi	从设备片选；
		GPIO_9			作为普通 I/O 使用；
1	89	**SS_3**	Dig	BiDi	从设备片选；
		GPIO_10			作为普通 I/O 使用；

表格 2-10 UART 口引脚

引脚数	Pin	名称	类型	方向	功能
1	75	**COMM_RXD**	Dig	BiDi	COMM 的 UART 口的接收数据口；
		GPIO_0			作为普通 I/O 使用；
1	76	**COMM_TXD**	Dig	BiDi	COMM 的 UART 口的发送数据口；
		GPIO_1			作为普通 I/O 使用；
1	77	**APP_RXD**	Dig	BiDi	APP 的 UART 口的接收数据口；
		GPIO_2			作为普通 I/O 使用；
1	78	**APP_TXD**	Dig	BiDi	APP 的 UART 口的发送数据口；
		GPIO_3			作为普通 I/O 使用；

表格 2-11 专用 GPIO 口引脚

引脚数	Pin	名称	类型	方向	功能
1	7	**GPIO_22**	Dig	BiDi	普通 I/O 口使用；
1	8	**GPIO_23**	Dig	BiDi	普通 I/O 口使用；

表格 2-12 复用 GPIO 口引脚

引脚数	Pin	名称	类型	方向	功能
1	93	**RTC_32KHZ**	Dig	BiDi	时钟 32.768KHz 信号输出；用于校准时钟精度；
		GPIO_13			普通 I/O 口使用；
1	94	**RTC_ALARM**	Dig	BiDi	RTC 报警信号输出；输出信号根据软件设计；
		GPIO_14			普通 I/O 口使用；
1	95	**COMM_T0**	Dig	BiDi	作为 COMM 的 T0 时钟输入；
		GPIO_15			普通 I/O 口使用；
1	96	**COMM_T1**	Dig	BiDi	作为 COMM 的 T1 时钟输入；
		GPIO_16			普通 I/O 口使用；
1	1	**COMM_T2**	Dig	BiDi	作为 COMM 的 T2 时钟输入；
		GPIO_17			普通 I/O 口使用；
1	2	**COMM_GATE0**	Dig	BiDi	作为 COMM 的 GATA0 输入；
		GPIO_18			普通 I/O 口使用；
1	3	**COMM_GATE1**	Dig	BiDi	作为 COMM 的 GATE1 输入；
		GPIO_19			普通 I/O 口使用；
1	5	**COMM_T2EX**	Dig	BiDi	作为 COMM 的 T2EX 输入；
		GPIO_20			普通 I/O 口使用；
1	6	**COMM_OSCEN**	Dig	BiDi	作为 COMM 的外部晶振输出；
		GPIO_21			普通 I/O 口使用；

表 格 2-13 I^2C 口引脚

引脚数	Pin	名称	类型	方向	功能
1	90	**SCL**	Dig	BiDi	I^2C 时钟信号线；
		GPIO_11			作为普通 I/O 使用；
1	92	**SDA**	Dig	BiDi	I^2C 数据线；
		GPIO_12			作为普通 I/O 使用；

2.2.4 载波模块引脚

表格 2-14 载波接收模块引脚

引脚数	Pin	名称	类型	方向	功能
1	97	**ADC_VCC3K**	Power	In	RX ADC 数字电源 3.3V ;对地跨个 0.1uf 滤波电容；
1	98	**ADC_GND3K**	Power	In	RX ADC 数字地；
1	99	**ADC_GNDA**	Power	In	RX ADC 模拟地；
1	100	**ADC_VCCA**	Power	In	RX ADC 模拟电源 3.3V （RX_3.3V）；

1	101	**ADC_VIP**	Analog	Out	内部 PGC 输出至 ADC 模拟信号输入正端，仅供检测用； 推荐接法外面加了滤波电路，具体参看用户手册参考电路。
1	102	**ADC_VIN**	Analog	Out	内部 PGC 输出至 ADC 模拟信号输入负端，仅供检测用；
1	103	**ADC_REFP**	Analog	BiDi	Rx ADC 上端参考电压；推荐接法对地 ADC_GNDA 跨个 0.1uF 电容；参考电压范围参看 DC 特性；
1	104	**ADC_VCM**	Analog	In	Rx ADC 共模输入基准电压；推荐接法对地 ADC_GNDA 跨个 0.1uF 电容；
1	105	**ADC_REFN**	Analog	In	Rx ADC 下端参考电压；推荐接法对地 ADC_GNDA 跨个 0.1uF 电容；参考电压范围参看 DC 特性；
1	106	**ADC_RSET**	Analog	BiDi	Rx ADC 电流参考；推荐接法对地 ADC_GNDA 跨个 12kΩ 电阻；
1	107	**PGC_GNDA**	Power	In	PGC（可编程增益模块）模拟地；
1	108	**PGC_VIN**	Analog	In	PGC 模拟电压输入，载波接收信号输入端；输入电压范围参看 DC 特性；
1	109	**PGC_VCM**	Analog	BiDi	PGC 模拟 DC 偏置电压端；推荐接法对地 ADC_GNDA 跨个 0.1uF 电容；直流偏置电压参看 DC 特性；
1	110	**PGC_VCCA**	Power	In	PGC 模拟电压源 3.3V； （RX_3.3V）；
1	72	**CAGC_GAIN_CNTL**	Dig	Out	自动增益（粗调）控制输出管脚；
1	71	**ZERO_CROSS**	Dig	In	过零检测引脚；

表格 2-15 载波发送模块引脚

引脚数	Pin	名称	类型	方向	功能
1	111	**DAC_GND3V**	Power	In	Tx DAC 数字地；
1	112	**DAC_VCC3V**	Power	In	TxDAC 数字电源 (TX_3.3V) Tx DAC 数字电源和数字地之间跨个 0.1 uF 电容；
1	113	**DAC_XDAC_OUT**	Analog	Out	Tx DAC 模拟信号输出； 电压范围参看 DC 特性；
1	114	**DAC_XVRB**	Analog	BiDi	Tx DAC 模拟参考电压 Vref-； 参考电压可由外部提供也可由内部提供；参考电压范围

					参看 DC 特性；
1	115	**DAC_XVRT**	Analog	BiDi	Tx DAC 模拟参考电压 Vref+ ;参考电压可由外部提供也可由内部提供；参考电压范围参看 DC 特性；

2.2.5 测试、调试引脚

表格 2-16 工厂成品测试引脚

引脚数	Pin	名称	类型	方向	功能
1	15	**TEST_CLK**	Dig	In	工厂成品测试，加个下拉电阻；
1	16	**SCAN_ENABLE**	Dig	In	工厂成品测试，加个下拉电阻；
1	17	**SCAN_TEST**	Dig	In	工厂成品测试，加个下拉电阻；
1	18	**TEST_MODE**	Dig	In	工厂成品测试，加个下拉电阻；
1	73	**SCAN_1**	Dig	In	工厂成品测试，加个下拉电阻；
1	74	**SCAN_2**	Dig	In	工厂成品测试，加个下拉电阻；

表格 2-17 DOCD 开发平台调试引脚

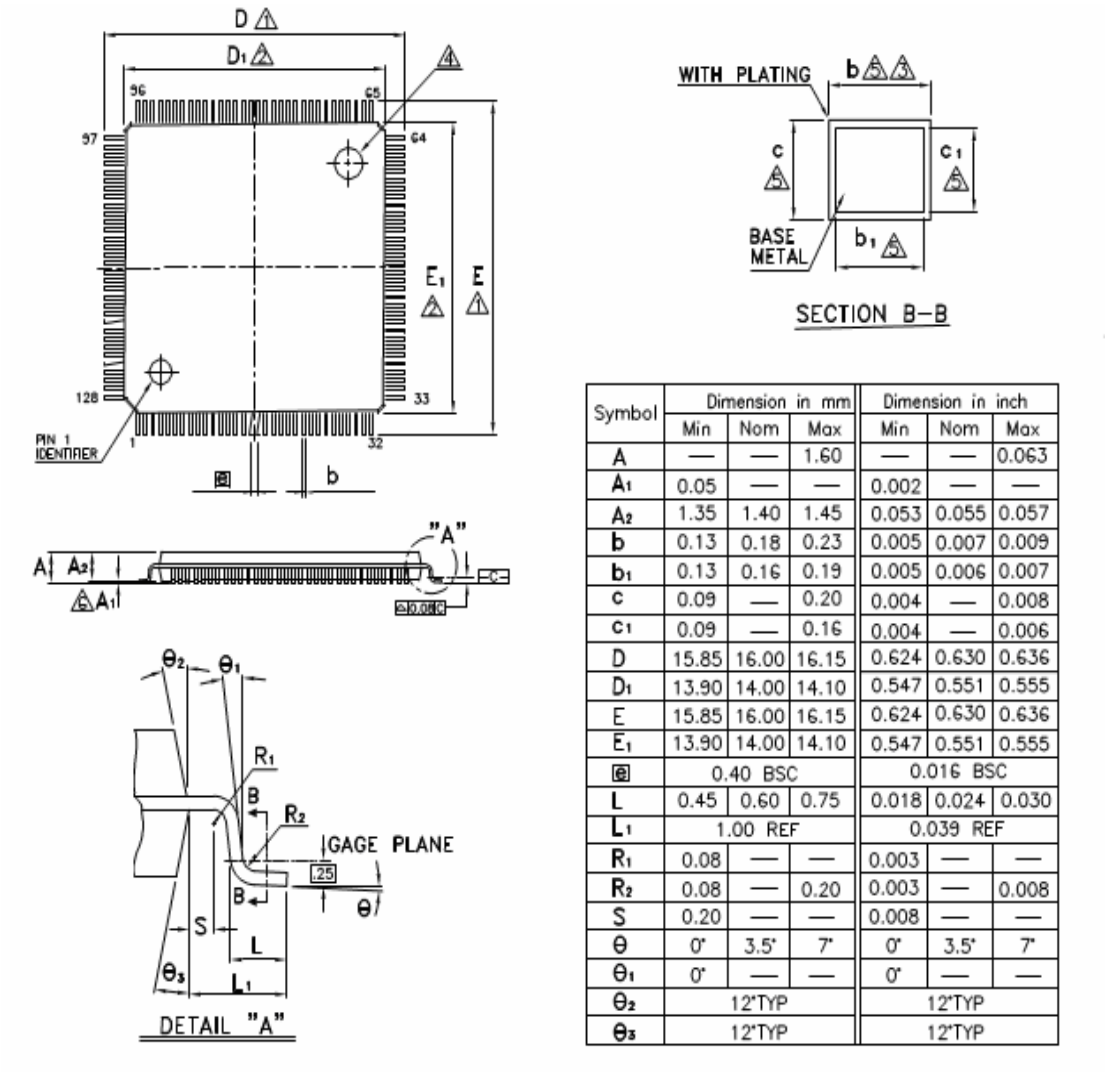
引脚数	Pin	名称	类型	方向	功能
1	58	**COMM_DOCDCLK**	Dig	Out	COMM_DOCD 调试时钟信号；
1	59	**COMM_DOCDDATAI**	Dig	In	COMM_DOCD 调试数据输入；
1	60	**COMM_DOCDDATAO**	Dig	Out	COMM_DOCD 调试数据输出；
1	61	**APP_DOCDCLK**	Dig	Out	APP_DOCD 调试时钟信号；
1	62	**APP_DOCDDATAI**	Dig	In	APP_DOCD 调试数据输入；
1	63	**APP_DOCDDATAO**	Dig	Out	APP_DOCD 调试数据输出；

表格 2-18 测试引脚

引脚数	Pin	名称	类型	方向	功能
1	122	**VC_SI_0**	Dig	In	芯片测试，加下拉电阻；
1	123	**VC_SI_1**	Dig	In	芯片测试，加下拉电阻；
1	125	**COMM_VC_SO_0**	Dig	Out	芯片测试，悬空；
1	126	**COMM_VC_SO_1**	Dig	Out	芯片测试，悬空；
1	127	**APP_VC_SO_0**	Dig	Out	芯片测试，悬空；
1	128	**APP_VC_SO_1**	Dig	Out	芯片测试，悬空；

2.3 封装参数

RISE3301—LQFP128 引脚封装参数请参见下图：



Symbol	Dimension in mm			Dimension in inch		
	Min	Nom	Max	Min	Nom	Max
A	—	—	1.60	—	—	0.063
A_1	0.05	—	—	0.002	—	—
A_2	1.35	1.40	1.45	0.053	0.055	0.057
b	0.13	0.18	0.23	0.005	0.007	0.009
b_1	0.13	0.16	0.19	0.005	0.006	0.007
c	0.09	—	0.20	0.004	—	0.008
c_1	0.09	—	0.16	0.004	—	0.006
D	15.85	16.00	16.15	0.624	0.630	0.636
D_1	13.90	14.00	14.10	0.547	0.551	0.555
E	15.85	16.00	16.15	0.624	0.630	0.636
E_1	13.90	14.00	14.10	0.547	0.551	0.555
e	0.40 BSC			0.016 BSC		
L	0.45	0.60	0.75	0.018	0.024	0.030
L_1	1.00 REF			0.039 REF		
R_1	0.08	—	—	0.003	—	—
R_2	0.08	—	0.20	0.003	—	0.008
S	0.20	—	—	0.008	—	—
θ	0°	3.5°	7°	0°	3.5°	7°
θ_1	0°	—	—	0°	—	—
θ_2	12°TYP			12°TYP		
θ_3	12°TYP			12°TYP		

图 2-2 封装参数图

注： 1.0.22um 超大规模数/模混合 CMOS 制造工艺。
2.128-pin LQFP 封装（14 x 14 x 1.4mm）；
3.工作温度范围（工业级标准）-40℃~+85℃；

第三章 存储器配置

片内嵌入两个增强型 8051；Memory 接口包括程序存储器和数据存储器两大部分，外扩两大存储空间通过并行 8 根数据线和 17 根地址线（128KB—0x00000~0x1ffff）来完成数据的传输和存储器空间的确定；程序存储器与数据存储器寻址空间在逻辑上是分开的，每个 MCU 程序存储器寻址空间各为 64KB，数据存储器寻址空间可达 16MB；。

3.1 程序存储器配置

1) 双 8 位高速 MCU 分别能外扩高达 64KB 的程序寻址空间；
2) 在 64KB 的程序寻址空间内，每个 MCU 各占用 2KB 的 SRAM 在片内作为高速的程序存储器（地址在高端）；地址分配如下：

处理器	片外 62K 地址	片内 2K 高速 SRAM 地址
COMM Processor	0x00000----0x0F7FF	0x0F800----0x0FFFF
APP Processor	0x10000----0x1F7FF	0x1F800----0x1FFFF

3) 硬件连接：除地址、数据线外的另外三根读写、使能、控制信号线 WE_N OE_N CE0_N；
4) 为使 MCU 与外挂的 Flash 在速度上达到匹配，可根据需要设置延时等待寄存器，具体过程见寄存器定义；
5) 程序存储器空间地址分配；

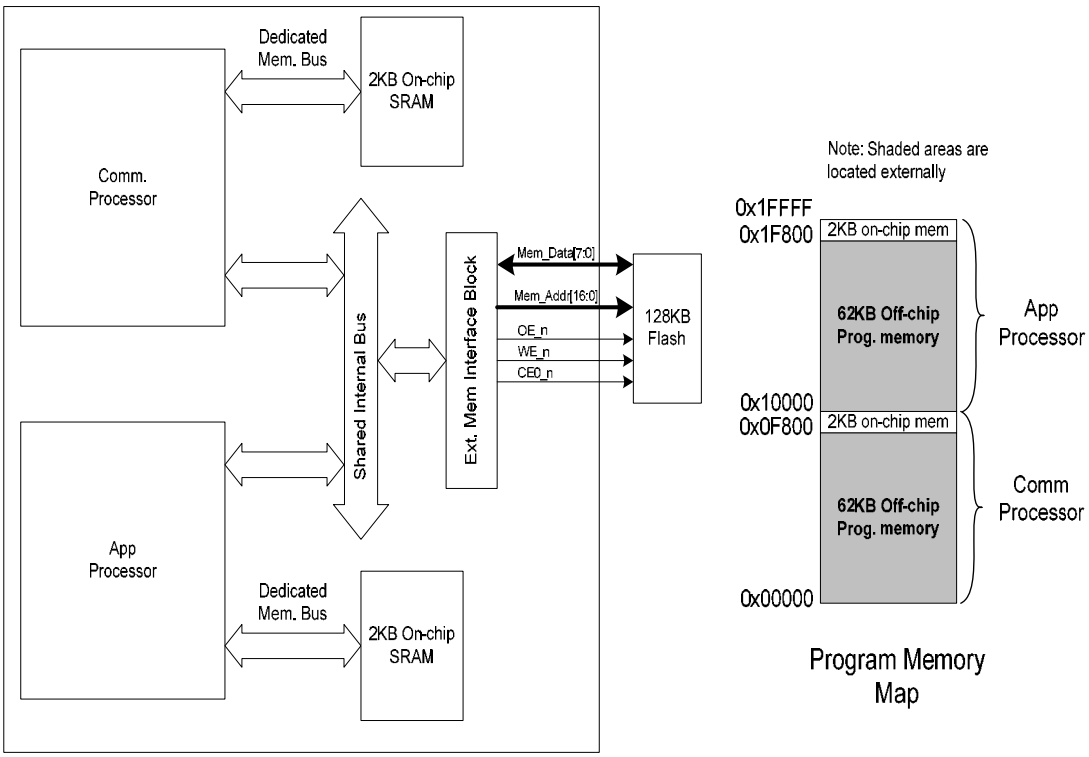


图 3-1 程序存储器空间地址分配图

3.2 数据存储器配置

1) 数据存储器寻址空间可达 16M，根据 CE0_N、CE1_N、CE2_N 引脚的控制确定寻址空间；
2) RISE3301 芯片片内有 10KB 的 SRAM（地址 0x000000~0x0027ff），可作为两个处理器数据暂存区；
3) RISE3301 芯片有三个外部片选端：CE0_N、CE1_N、CE2_N。CE1_N 在处理器访问第一高端 128KB 数据存储器块时有效（0xFE0000~0xFFFFFF），CE2_N 在处理器访问第二高端 128KB 数据存储器块时有效（0xFC0000~0xFDFFFF）；另外值得注意的是：数据存储器地址（0x020000~0x03ffff ）是 CE0_N 程序存储器（128K）映射在数据区的地址；所以用户在访问数据存储器地址（0x020000~0x03ffff）时，其实是在访问 CE0_N 程序存储器对应的地址。处理器以写地址范围在 0x020000~0x02FFFF、0x030000~0x03FFFF 的数据存储器的方式分别刷新放在 FLASH 中的 COMM、APP 处理器的程序代码；数据存储器对应分配表如下：

表格 3-1 数据存储器对应分配表

管脚使能	地址分配	功能
CE0_N	可连接程序存储模块或数据存储模块	选择程序运行空间片选控制；
CE1_N	0xFE0000~0xFFFFFF	可作为数据缓冲区、变量使用；
CE2_N	0xFC0000~0xFDFFFF	可作为数据缓冲区、变量使用；

4) 数据存储器空间地址分配；

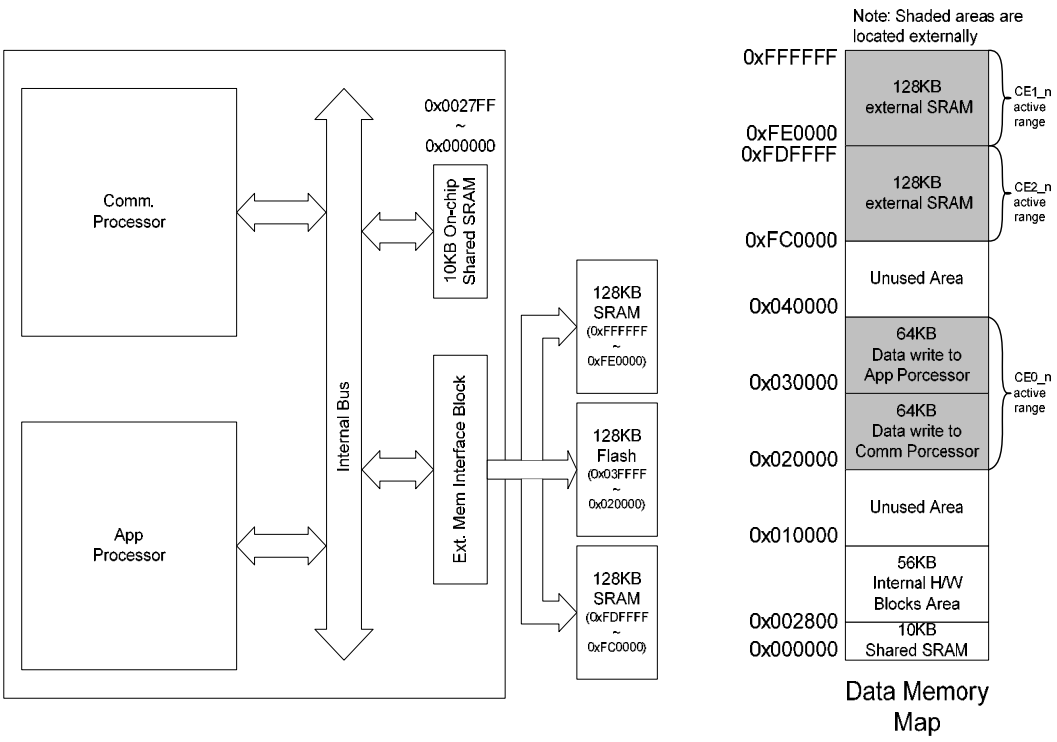


图 3-2 数据存储器空间地址分配图

5) 假如用户要节省从 Flash 中读取指令所花费的时间，可通过下列步骤将程序搬移到CE1_N所指向的SRAM，通过使能EBIM控制寄存器的位CE1_SELECT，从而实现用户从 CE1_N 所指向的 SRAM 中读取指令。

具体实现步骤如下：

- 通过软件读取数据存储器地址范围 0x020000~0x03ffff 中的内容，然后将其对应移入 CE1_N 所指向的 SRAM 中；
- 置 EBIM 控制寄存器的 CE1_SELECT 位为 1；
- 下一条指令就从 SRAM 中读取。整个应用程序运行在 CE1_N 所指向的数据存储器。

6) 硬件连接：指除地址、数据线外的另外三根读写、使能、地址配置信号线：WE_N、OE_N 与 CE0_N（或 CE1_N、CE2_N）；

7) 寄存器定义

表格 3-2 外部总线接口寄存器

外部总线接口寄存器 (Base Addr: 0x00d000)

偏移地址	寄存器名称	寄存器符号	类型	复位状态
0000	CE0、CE1 延时等待寄存器	WTST01_COUNTERS	R/W	0xff
0001	CE2 延时等待寄存器	WTST2_COUNTER	R/W	0xff
0002	邮件盒寄存器	MBX_HANDSHAKE	R/W	0x01
0003	预留	---	---	---
0004	EBIM 控制寄存器	EBIM_C	R/W	0x00

表格 3-3 延时等待寄存器

延时等待寄存器			功能描述
WTST01_COUNTERS	Bit0~Bit3	WTST_C0	连接 CE0_n 的芯片所对应的延时等待寄存器； 例如：寄存器的值为 N；则延时等待时间则为 N*T；数据为 0 时代表最快存取时间，即 75ns； T---Clk_bus 为 clkrun/2； (默认 80Mhz 的工作频率下 T=25ns)
	Bit4~Bit7	WTST_C1	CE1_n 延时等待周期；同上；
WTST2_COUNTER	Bit0~Bit3	WTST_C2	CE2_n 延时等待周期；同上；
	Bit4~Bit7		预留

表格 3-4 EBIM 控制寄存器

EBIM 控制寄存器			功能描述
EBIM_C	Bit0~Bit2		预留
	Bit3	CE1_SELECT	该标志置 1，下一条指令将从 SRAM 中读取
	Bit4~Bit7		预留

8) Memory 接口读写时序

表格 3-5 Memory 接口读写时序表

符号	参数	最小	典型	最大	单位
T1	CEx_N active low to External Address valid			0.4	ns
T2	CEx_N active low to OE_N valid			13	ns
T3	External Data Setup time	25			ns
T4	External Data hold time	0			ns
T5	OE_N inactive to CEx_N inactive			12.5	ns
T6	External Address hold time	25			ns
T7	Memory Access Time	75		450	ns
T8	CEx_N active low to WE_N and External Data valid	12.5			ns
T9	WE_N inactive to CEx_N inactive			12.5	ns
T10	Ext_Data hold time to WE_N inactive	10			ns

注：Memory 读写时序由存储在 CE0_N， CE1_N 以及 CE2_N 等待状态寄存器的等待状态数量决定。

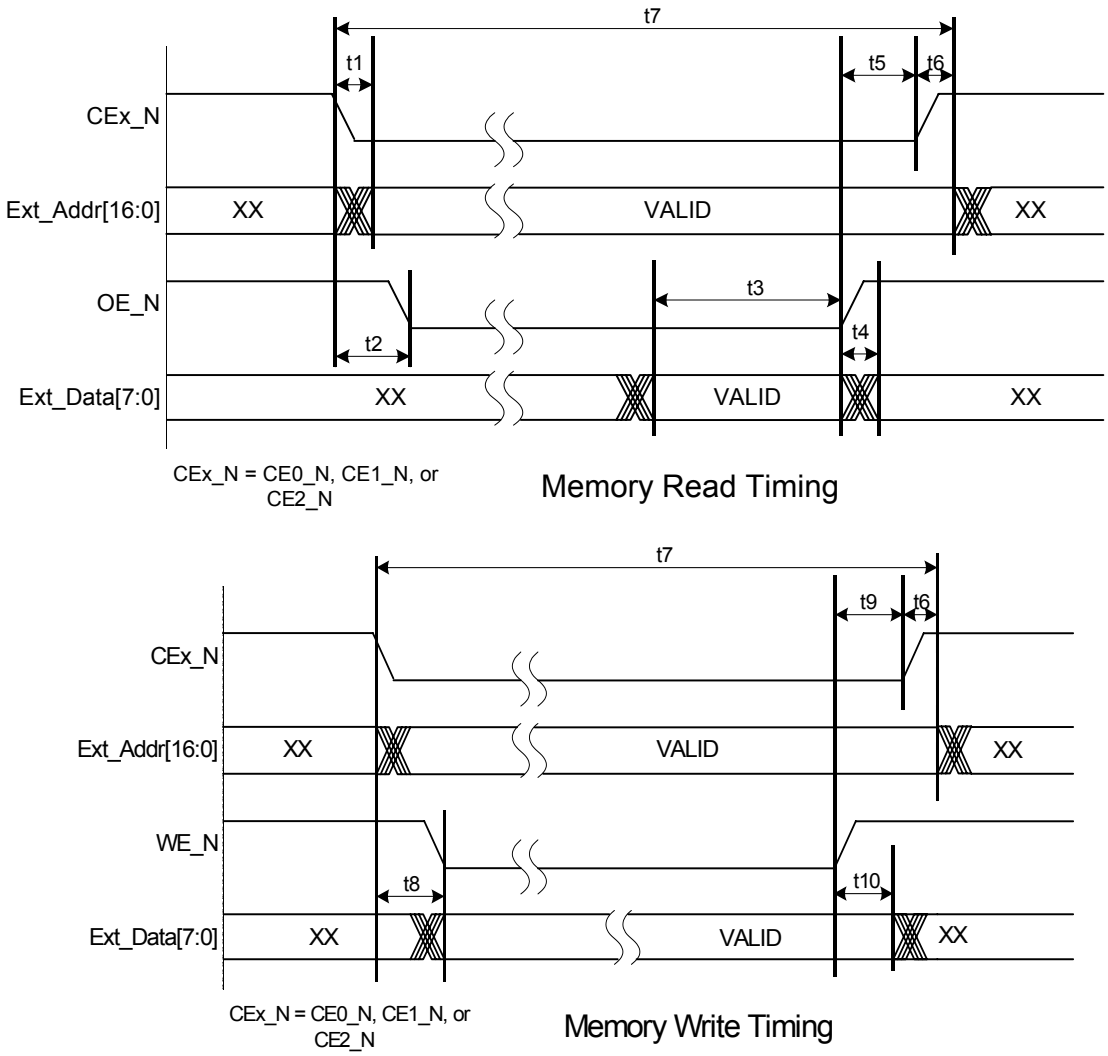


图 3-3 Memory 读写时序图

3.3 片内数据存储器配置

片内各有 256 个 RAM 单元，地址范围为 00H~FFH（256Byte）。其中前 128 个单元是一个多功能数据存储器，有数据存储、通用工作寄存器、堆栈、位地址等空间；后 128 个单元作为专用寄存器（SFR）。

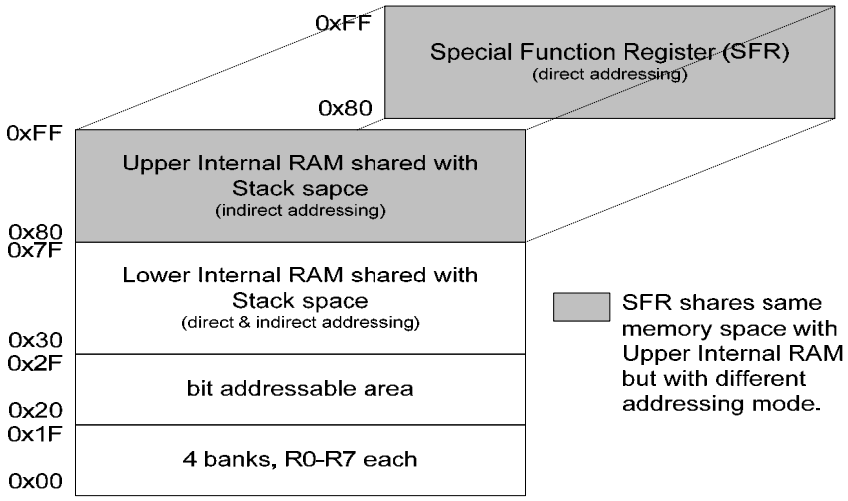


图 3-4 片内 256 个 RAM 单元分配

特殊功能寄存器（0x80-0xff）地址定义：

表格 3-6 特殊功能寄存器地址定义表

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
0xF8	EIP								0xFF
0xF0	B								0xF7
0xE8	EIE		**MXAX**						0xEF
0xE0	ACC								0xE7
0xD8	WDCON								0xDF
0xD0	PSW								0xD7
0xC8	T2CON		RLDL	RLDH	TL2	TH2			0xCF
0xC0						STATUS		TA	0xC7
0xB8	IP								0xBF
0xB0	P3								0xB7
0xA8	IE								0xAF
0xA0	P2								0xA7
0x98	SCON0	SBUF0							0x9F
0x90	P1	EIF	**WTST**	**DPX0**		**DPX1**			0x97
0x88	TCON	TMOD	TL0	TL1	TH0	TH1	**CKCON**		0x8F
0x80	P0	SP	DPL0	DPH0	**DPL1**	**DPH1**	**DPS**	**PCON**	0x87
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
黑体:新增寄存器									
常规:标准 8051 寄存器									
阴影:外围控制寄存器									

PSW (0xD0)

7	6	5	4	3	2	1	0	Reset
CY	AC	F0	RS1	RS0	OV	F1	P	0x00

CY - 进位标志
AC - 辅助进、借
F0 - 用户标志位，由用户决定该标志位
RS[1:0] 00 -Bank 0， 地址范围 0x00-0x07
01 -Bank 1， 地址范围 0x08-0x0F
10 -Bank 2， 地址范围 0x10-0x17
11 -Bank 3， 地址范围 0x18-0x1F
OV - 溢出标志位
F1 - 用户标志位，由用户决定该标志位
P - 奇偶校验位

中断控制寄存器
IE (0xA8)

7	6	5	4	3	2	1	0	Reset
EA	-	ET2	ES0	ET1	EX1	ET0	EX0	0x00

EA - 全局中断开关
EX0 - 外部中断INT0使能位
ET0 - Timer 0 中断使能位
EX1 - 外部中断INT1使能位
ET1 - Timer 1 中断使能位
ES0 - UART0 口中断使能位
ET2 - Timer 2 中断使能位

EIE (0xE8)

7	6	5	4	3	2	1	0	Reset
-	-	-	EWDI	EINT5	EINT4	EINT3	EINT2	0x00

EINT2 -外部中断INT2使能位
EINT3 -外部中断INT3使能位
EINT4 -外部中断INT4使能位
EINT5 -外部中断INT5使能位
EWDI - WATCHDOG中断使能位

中断优先级控制寄存器
IP(0xB8H)

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

BIT7-保留
BIT6-保留
PT2（BIT5）-定时器 2 中断优先（8052）
PS （BIT4）-串行口中断优先
PT1（BIT3）-定时器 1 中断优先
PX1（BIT2）-外中断 INT1 中断优先
PT0（BIT1）-定时器 0 中断优先
PX0（BIT0）-外部中断 INT0 的中断优先
具体 SFR 的详细定义请参照 51 的相关资料！

3.4 片内 SRAM 的分配

片内共有 14KB SRAM，具体分配如下

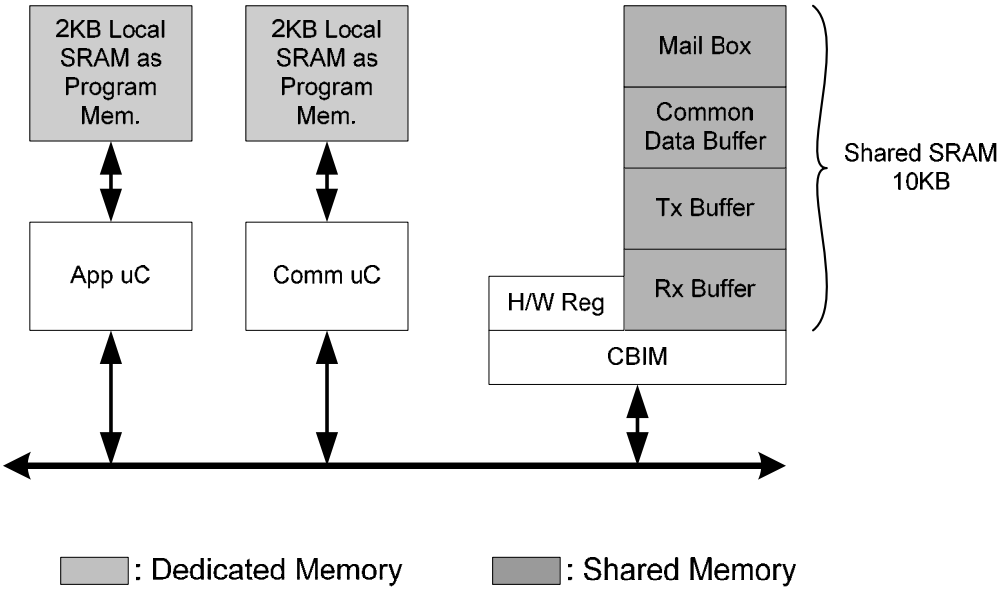


图 3-5 片内 SRAM 分配图

从图中可以看出 APP 与 COMM 各有 2KB 专用 SRAM 作为片内高速程序存储器，供底层软件使用；共享的 10KB SRAM 作为网络数据缓冲，由四部分硬件数据缓冲区所组成，分别是 Rx Buffer、Tx Buffer、Common Data Buffer、以及 Mail Box。具体定义如下：

1) Rx Buffer、Tx Buffer 各 1KB 的 SRAM；
2) Common Data Buffer 为 COMM 与 APP 通用数据缓冲区，
3) Mail Box 应用于内嵌 COMM 与 APP 之间的数据交换缓冲；

给用户预留的具体存储变量空间请参考用户手册中设计方案的举例说明。

3.5 Mail Box 功能说明

Mail Box 主要用于 RISE3301 芯片内 COMM 与 APP 之间的数据交换传输，Mail Box 在中间起到透明传输的作用。

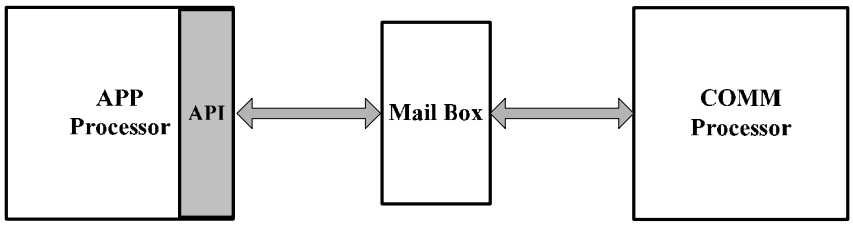


图 3-6 Mail Box 功能图

COMM 端底层软件负责网络管理事务，APP 端负责用户应用开发；APP 与 COMM 之间的数据交换是通过 Mail Box 的透明传输实现的，用户不需要关注 Mail Box 的具体操作，只需调用 API 接口函数即可实现对应的功能。

第四章 电力载波发送和接收模块

4.1 描述

电力载波发送/接收信号传输由底层硬件支持，采用 BPSK 调制解调技术结合先进的 FIR 和 IIR 数字滤波以及高级的 DSP 处理技术，使用专门的硬件来检测（检测什么东西），并在接收链上消除脉冲噪音；用多级数字和模拟滤波器可有效消除多点频噪音，采用高级信道获取算法可保证即使在信号低于噪音的情况下，也能够获取信号。RISE3301 芯片采用自动增益控制（AGC），AGC 划分成模拟和数字域两部分，使 RISE3301 芯片能够在很宽的范围调整接收信号电平。此外，RISE3301 芯片的 COMM 已嵌入并固化了 EIA709.1 协议的协议栈和系统自动路由算法模块，从而使与载波网络管理相关的所有事务都交由协议栈来完成；涉及到载波通讯的数据则透明传输给 APP，由 APP 完成数据的解析；COMM 与 APP 之间通过 API 接口函数实现功能调用。

4.2 RISE3301 载波发送模块的原理及特性

4.2.1 原理

待发送的二进制数据通过 Packet Assemble 打包成数据帧格式，由 BPSK 调制成数字信号，经过数字滤波器以及 10 位 D/A 转换成模拟信号输出。这就是 RISE3301 载波发送模块的原理（参见下图）。

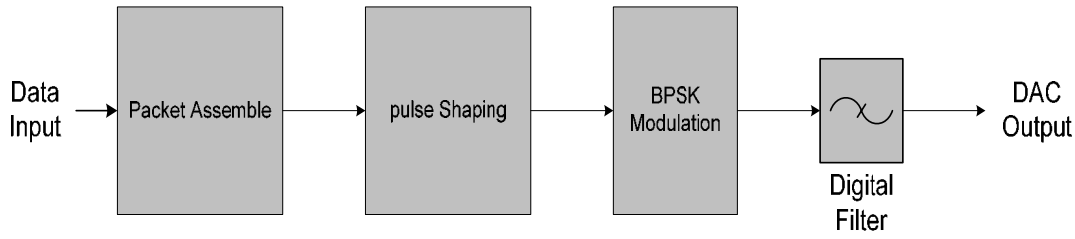


图 4-1 载波发送底层硬件原理图

4.2.2 特性

载波发送模块的特性有：

- 电力线数据通讯波特率可达 5.5kbps；
- 中心频率衰减 3dB 后的载波信号带宽为 5KHz、衰减 20dB 后的载波信号带宽为 10KHz；
- 接收及发送数据处理（DSP）模块，按 EIA709.1、EIA709.2、及 EN50065-1 国际标准设计。

4.3 RISE3301 载波接收模块的原理及特性

4.3.1 原理

载波信号经过外围滤波电路进入 RISE3301 芯片内部的 PGC 增益控制，由 10 位的 A/D 转换成数字信号，数字信号经过数字滤波电路、噪声抑制电路、波形匹配滤波器以及 AGC 自动增益控制电路后进入 Channel Acquisition 捕捉器，被 Channel Acquisition 捕捉到的数据帧解调为二进制数据格式。

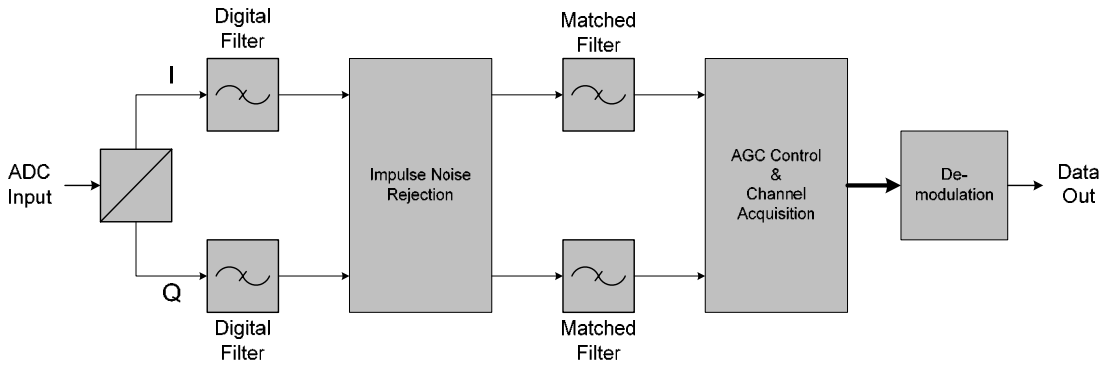


图 4-2 载波接收底层硬件原理图

4.3.2 特性

载波接收模块的特性有：

- 脉冲干扰抑制；
- 快速信道自适应；快速自动增益控制；
- 取样时间误差自动纠正（可达 200 ppm 以上）；
- 接收信号质量监控，监控内容多样，可包括：信号功率、干扰强度、误码率等。

第五章 系统时钟

5.1 描述

系统时钟信号由 PLL 倍频得到 ;经过 PLL 倍频后所得到的 OSC 又经过分频得到三种时钟频率供内部使用；默认外部时钟频率为 10MHz。

5.2 系统时钟介绍

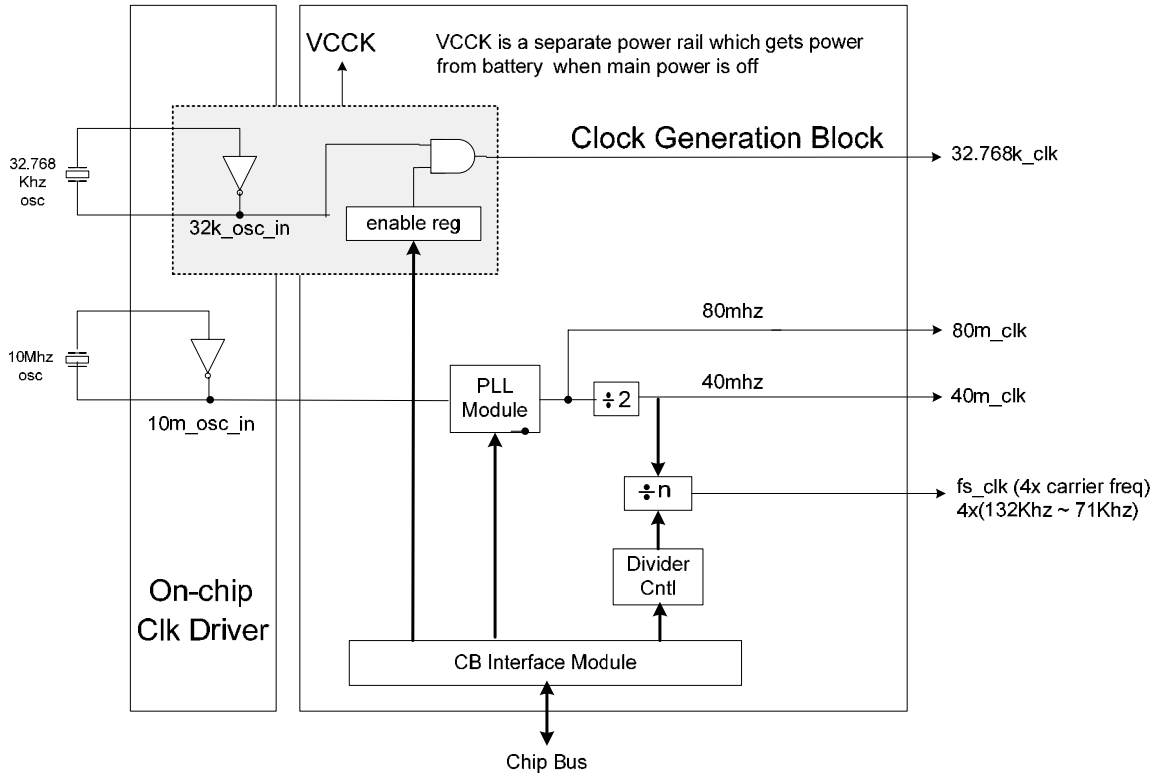


图 5-1 系统时钟图

表格 5-1 时钟信号表

时钟信号频率	使用对象
80MHz	微处理器时钟；最高工作频率（也是默认值）为 80MHz，具体系统设置的工作频率以软件说明为准；
40 MHz	Tx、Rx 和其它外围设备；
Fs_clk	载波取样频率源；底层软件根据配置所得到四个频点分别为：75KHz、86KHz、115KHz、132KHz；

*10MHz 外部输入晶振，PLL 默认倍率为 8 倍频

第六章 中 断

RISE3301 的中断功能非常强大，有 15 个可选逻辑硬件配置中断源，其中包括 6 个外部中断以及 I^2C、SPI、TX、RX 中断，内部 Timer 以及 UART 口为 8051 本身中断；有关具体外部中断的配置方法以及底层占用情况请参看用户手册。

第七章 GPIO 口

7.1 GPIO 口描述

RISE3301 共有 24 个 GPIO 口，其中复用口有 22 个，分别作为 I^2C、SPI、UART 口等多功能口；复用口的第二功能由寄存器 P3 口定义实现；由于 RISE3301 芯片有两个内核，每个内核各带 256 字节的数据寄存器， GPIO 口的复用功能实现是由 COMM 端的 P3 寄存器控制。每个 GPIO 口都为双向，并且内部带有上拉。GPIO 口可通过方向寄存器分别定义为输入输出口，‘ 1 ’为输出，‘ 0 ’为输入。

7.2 复用口介绍

其中22个复用GPIO口具体作为普通I/O口还是第二功能口使用由P3口定义决定，(P3 寄存器的配置是通过 APP 调用 API 接口函数实现的，由 COMM 完成 P3 寄存器的配置)。P3 (0xB0)寄存器在复位后的状态是 0xff，GPIO 复用口默认为普通 I/O 口。
复用口介绍如下：

表格 7-1 复用口表

Pin	Default Function	Alternate Function
76	GPIO_0	COMM_RXD[-1]
77	GPIO_1	COMM_TXD[-1]
78	GPIO_2	APP_RXD[-2]
79	GPIO_3	APP_TXD[-2]
81	GPIO_4	SCLK[-3]
82	GPIO_5	MOSI[-3]
83	GPIO_6	MISO[-3]
84	GPIO_7	SS_0[-3]
85	GPIO_8	SS_1[-3]
86	GPIO_9	SS_2[-3]
89	GPIO_10	SS_3[-3]
91	GPIO_11	SCL[-4]
92	GPIO_12	SDA[-4]
93	GPIO_13	RTC_32KHZ[-5]
94	GPIO_14	RTC_ALARM[-6]
95	GPIO_15	COMM_T0[-7]
96	GPIO_16	COMM_T1[-7]
1	GPIO_17	COMM_T2[-7]
2	GPIO_18	COMM_GATE0[-7]
3	GPIO_19	COMM_GATE1[-7]
5	GPIO_20	COMM_T2EX[-7]
6	GPIO_21	COMM_OSCEN[-9]
7	GPIO_22	
8	GPIO_23	

*表格 7-1 的 Alternate Function 列中右上标（-1、-2…）的定义见表格 7-2

表格 7-2 上标定义

	对应位	定义说明
- 1	P3[0]	0---作为 COMM 的 UART 口使用 1----作为普通 I/O 口使用
- 2	P3[1]	0---作为 APP 的 UART 口使用 1----作为普通 I/O 口使用
- 3	P3[2]	0---作为复用口 SPI 使用 1----作为普通 I/O 口使用
- 4	P3[3]	0---作为复用口 IIC 使用 1----作为普通 I/O 口使用
- 5	P3[4]	0---作为复用口 RTC_32KHz 使用 1----作为普通 I/O 口使用
- 6	P3[5]	0---作为复用口 RTC_ALARM 使用 1----作为普通 I/O 口使用
- 7		这些复用功能默认作为 COMM 口使用； 具体使能由内部控制寄存器配置。
- 9	P3[6]	0----作为 COMM 的复用口使用 1----作为普通 I/O 口使用

7.3 GPIO 口寄存器

表格 7-3 GPIO 口寄存器

GPIO 端口寄存器（BASE ADDR：0x00c000）				
偏移地址	寄存器名称	寄存器符号	类型	复位状态
0000	GPIO 口数据寄存器 1	GPIO_DAT_1	R/W	----
0001	GPIO 口数据寄存器 2	GPIO_DAT_2	R/W	----
0002	GPIO 口数据寄存器 3	GPIO_DAT_3	R/W	----
0003	GPIO 口方向寄存器 1(GPIO_0~ GPIO_7)	GPIO_EN_1	R/W	0x00
0004	GPIO 口方向寄存器 2(GPIO_8~ GPIO_15)	GPIO_EN_2	R/W	0x00
0005	GPIO 口方向寄存器 3(GPIO_16~ GPIO_23)	GPIO_EN_3	R/W	0x00
0006~ 0001d	GPIO 端口位操作寄存器 0 ~ 23	GPIO_DAT_Bx	R/W	----

*GPIO_DAT_1、GPIO_DAT_2、GPIO_DAT_3 的状态值为 GPIO 口口线状态值；

GPIO 端口位操作寄存器 0 ~ 23

地址: C006 ~ C01D 类型: R

符号名称: GPIO_DAT_Bx (x = 0 ~ 23)

位	（字段名称）	功能描述	复位值
7:0		全部 24 个 GPIO 口位操作寄存器地址为：C006 ~ C01D. 每一个寄存器对应一个口.如 地址 C006 对应 GPIO 口 0，地址 C01D 对应 GPIO 口 23。 操作每个寄存器的数据，对 GPIO 口有效的仅仅是一位，对应的有效位如下：	R/W

位操作寄存器	有效位
Register-0 (C006)	bit-0
Register-1 (C007)	bit-1
Register-2 (C008)	bit-2
Register-3 (C009)	bit-3
Register-4 (C00a)	bit-4
Register-5 (C00b)	bit-5
Register-6 (C00c)	bit-6
Register-7 (C00d)	bit-7
Register-8 (C00e)	bit-0
Register-9 (C00f)	bit-1
Register-10 (C010)	bit-2
Register-11 (C011)	bit-3
Register-12 (C012)	bit-4
Register-13 (C013)	bit-5
Register-14 (C014)	bit-6
Register-15 (C015)	bit-7
Register-16 (C016)	bit-0
Register-17 (C017)	bit-1
Register-18 (C018)	bit-2
Register-19 (C019)	bit-3
Register-20 (C01a)	bit-4
Register-21 (C01b)	bit-5
Register-22 (C01c)	bit-6
Register-23 (C01d)	bit-7

注：GPIO 口的具体使用情况请详见用户手册的硬件资源配置章节。

第八章 通讯口/Timer 功能

8.1 UART 口

8.1.1 UART 口描述

RISE3301 内嵌的两个内核各带一个全双工 UART 口。 UART 可配置为全双工异步系统，实现与 CRT 终端和个人计算机等外围设备进行通讯；也可配置为半双工同步系统，实现与 A/D 或 D/A 集成电路及串行通讯等外围器件进行通讯；

8.1.2 UART 口工作模式

全双工异步模式，波特率换算的频率为 clk_run/12(80MHz/12)
8 位同步模式，波特率由定时器控制
9 位同步模式，波特率换算的频率为 clk_run/32 或 clk_run/64
9 位同步模式，波特率由定时器控制

UART 口的硬件连接：
COMM 数据接收口------COMM_RXD(GPIO_0)
COMM 数据发送口------COMM_TXD(GPIO_1)
APP 数据接收口------APP_RXD(GPIO_2)
APP 数据发送口------ APP_TXD(GPIO_3)

如若将 I/O 口配置为 UART 口使用，需定义寄存器：
1.P3 寄存器的 BIT0、BIT1 配置为 0，实现 COMM、APP 的 UART 口功能；
2.UART 口相关连的寄存器 SCON0、Data Buffer (SBUF0)， Serial-port Configuration (SCON0)， Power Configuration (PCON)， Interrupt Enable (IE) and Interrupt Priority (IP)。
3.寄存器介绍以及工作模式中的波特率的生成详见 FR8052 资料。

8.2 I2C 通讯口

8.2.1 I^2C 通讯口描述

I^2C 是 RISE3301 的两线制串行接口。通过 I^2C 接口可实现多片主、从机的互连，也可以实现 MCU 与其他具有 I^2C 串行接口的多个芯片互连，简化系统的电路设计。

I^2C 对应的两个引脚分别为：
Serial clock， SCL------ GPIO_11/SCL
Serial data， SDA ------ GPIO_12/SDA
用户必须通过 P3（BIT3）将这些引脚配置为专用口。

8.2.2 原理框图及工作模式

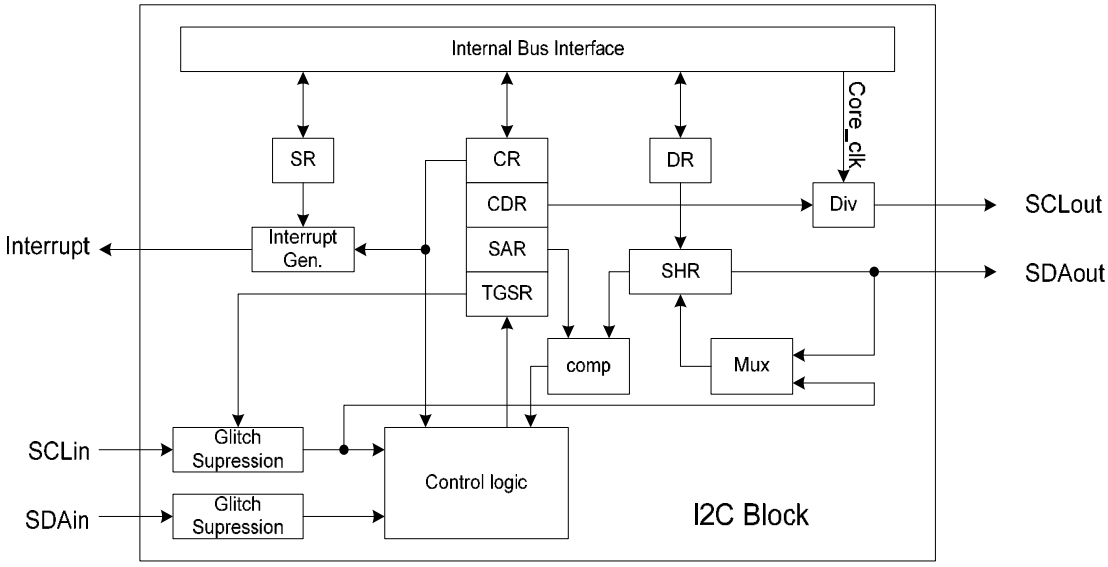


图 8-1 I^2C 原理图

RISE3301 集成硬件 I^2C Core 和 I^2C 总线模块，I^2C 模块完全可以实现所有主控和从动模式（包括全局呼叫支持）；并且通过硬件提供启动位中断和停止位中断以确定空闲总线（多主控模式）；I^2C 实现标准模式规范，以及 7 位寻址和 10 位寻址。I^2C 工作频率由 40MHz 经过分频得到。

8.2.3 工作寄存器

工作寄存器可分为：I^2C 控制寄存器、状态寄存器、从设备地址寄存器、串行时钟分频器、状态映射寄存器等，以下分步介绍各个寄存器的性能。

表格 8-1 I^2C 模块寄存器综述（基本地址: 0x005000)

偏移地址	寄存器名称	寄存器符号	类型	Reset
0000	I^2C 控制寄存器 1	I2C_CR_1	R/W	0x00
0001	I^2C 控制寄存器 2	I2C_CR_2	R/W	0x00
0002	I^2C 控制寄存器 3	I2C_CR_3	R/W	0x00
0004	I^2C 状态寄存器 1	I2C_SR_1	R/RC	0x00
0005	I^2C 状态寄存器 2	I2C_SR_2	R/RC	0x00
0008	I^2C 时钟分频寄存器 1	I2C_CDR_1	R/W	0x00
0009	I^2C 时钟分频寄存器 2	I2C_CDR_2	R/W	0x00
000c	I^2C 数据寄存器	I2C_DR	R/W	0x00
0010	I^2C 从设备地址寄存器 1	I2C_SAR_1	R/W	0x00
0011	I^2C 从设备地址寄存器 2	I2C_SAR_2	R/W	0x00
0013	I^2C 从设备地址寄存器 3	I2C_SAR_3	R/W	0x00
0014	I^2C Hold Time/Glitch Suppression	I2C_TGSR_	R/W	0x00

偏移地址	寄存器名称	寄存器符号	类型	Reset
	Register_1	1		
0015	I²C Hold Time/Glitch Suppression Register_2	I2C_TGSR_2	R/W	0x00
0020	I²C 状态映射寄存器 1	I2C_SSR_1	R	0x00
0021	I²C 状态映射寄存器 2	I2C_SSR_2	R	0x00

8.2.3.1 I²C控制寄存器

控制寄存器位定义详见表格 8-2：

表格 8-2 控制寄存器位定义

控制寄存器			功能描述
I2C_CR_1	Bit0	I2C_RST	I²C Core 复位控制位，硬件复位后该标志位在两个机器周期后自动被清零； “1” = 复位使能 “0” = 不复位
	Bit1	I2C_EN	I²C 总线接口控制位； “1” = 复位使能 “0” = 不复位
	Bit2	SCL_EN	主工作模式下： I²C 时钟信号使能位 “1” = 时钟信号使能输出 “0” = 关掉 从工作模式置 “ 0 ”；
	Bit3	GC_EN	从工作模式全局呼叫使能； “1” = 使能 “0” = 不使能
	Bit4	START	主工作模式下发送数据起始位 “ S ” 使能控制位； “1” = 发送数据起始位 “ S ” “0” = 空闲
	Bit5	STOP	主工作模式下发送数据结束位“ P ”以表示数据发送完毕； “1” = 发送数据结束位 “0” = 不发送
	Bit6	ACK / NACK	主、从接收数据模式的应答信号控制位； “1”: 无应答发送 “0”: 有应答发送
	Bit7	TB_EN	数据发送接收使能控制位； “1” = 发送接收数据使能 “0” = 插入等待状态
I²C _CR_2	Bit0	DTI_EN	I²C 数据发送中断控制位； “1” = 打开数据发送中断 “0” = 关掉数据发送中断
	Bit1	DRI_EN	I²C 数据接收中断控制位； “1” = 打开数据接收中断

			“0” = 关掉数据接收中断
	Bit2	BERRI_EN	主工作模式下，数据发送完毕后应答位检测中断控制位； “1” = 数据发送完毕后未接收到应答标志 “0” = 关掉应答位检测中断控制
	Bit3	STOPI_EN	停止位中断检测控制位； “1” = 打开停止位中断 “0” = 关掉停止位中断
	Bit4	SAMI_EN	从工作模式下，从设备地址或全局呼叫地址（GC_EN=1）匹配中断控制位； “1” = 打开中断 “0” = 关掉
	Bit5	ALI_EN	主工作模式下，总线发生冲突； “1” = 打开总线冲突中断 “0” = 关掉总线冲突中断
	Bit6	STARTI_EN	起始位中断控制位； “1” = 打开中断 “0” = 关掉中断
	Bit7	SCL_LOW	特殊场合使用； “1” = SCL 无时钟信号输出，该电平拉低 “0” = 工作模式 常规使用该位置 0。
I^2C _CR_3	Bit0	SDA_LOW	特殊场合使用； “1” = SDA 无数据信号输出，该电平拉低 “0” = 工作模式 常规使用该位置 0。
	Bit1	TEST_BIT	测试位，该标志位置 0；
	Bit2~ Bit7		预留；

8.2.3.2 状态寄存器(SR)

状态寄存器反应当前工作状态，当发生读状态寄存器（读任何一个）命令时则硬件自动对该寄存器某些位（S）清零。状态寄存器的值自动被导入状态映射寄存器（SSR），编程者可以从状态映射寄存器（SSR）中获取相关状态位。

表格 8-3 状态寄存器

状态寄存器			功能描述
I^2C _SR_1	Bit0	RW	I^2C 处于主接收或从发送工作状态标志； “1”: 工作在主接收或从发送工作模式；
	Bit1	ACK	主、从模式的应答信号状态位； “1”: 无应答标志；
	Bit2	I2CB	I^2C Core 忙状态标志位，表示 I2C 工作在数据传输过程； “1”: 工作在忙状态；
	Bit3	BB	I^2C 总线忙状态标志位； “1”:忙状态；此时 I2C Core 不一定处于忙状态；

			I²C Core 和 I²C 总线两部分独立；
	Bit4	DT	数据寄存器（DR)发送数据标志位；(S)；
	Bit5	DR	数据寄存器（DR)接收数据标志位；(S)；
	Bit6	BERR	主工作模式下,数据发送完毕后应答位(no_ack)检测状态位；(S)；
	Bit7	STOP	停止状态标志位；(S)；
I²C _SR_2	Bit0	SAM	从工作模式下：从设备地址匹配标志；(S)；
	Bit1	GC	从工作模式下： 全局呼叫地址匹配标志；(S)；
	Bit2	AL	主工作模式下： 总线冲突状态标志位；(S)；
	Bit3	START	起始位状态标志位；(S)；
	Bit4~ Bit7		预留；

8.2.3.3 从设备地址寄存器(SAR)

从工作模式下，从设备地址定义。

表格 8-4 从设备地址寄存器

从模式设备地址			功能描述
I²C _SAR_1	Bit0~ Bit6	SAR	地址寄存器位定义，当 EN10=1 时，Bit0~Bit6 为 10 位从设备地址的低 7 位地址； 当 EN10=0 时,从设备地址为 7 位来处理；
	Bit7	SAR	地址寄存器位定义，当 EN10=1 时，作为 10 位从设备地址用；
I²C _SAR_2	Bit0~Bit1	SAR	地址寄存器位定义，当 EN10=1 时，作为 10 位从设备地址用；
	Bit2~ Bit7		预留；
I²C _SAR_3	Bit0~ Bit6		预留；
	Bit7	EN10	从设备地址使能位； “1” = 10 位地址工作模式 “0” = 7 位地址工作模式

8.2.3.4 串行时钟分频器(CDR)

通过串行时钟分频器换算得到 I²C 的时钟工作频率。

表格 8-5 时钟分频寄存器功能

时钟分频寄存器	功能描述
I²C _CDR_1	串行时钟频率计算方法： SCLout = 40M / (2*COUNT – GSR + 1) COUNT 为定义 10 位 BIT，I2C_CDR_1 为 COUNT 的低 8 位定义； （10M 外部晶振，8 倍频为参考）
I²C _CDR_2	串行时钟频率计算方法： SCLout = 40M / (2*COUNT – GSR + 1) COUNT 为定义 10 位 BIT，I²C _CDR_1 为 COUNT 的低 8

	位定义； I^2C _CDR_2 中 Bit0-bit1 为 COUNT 的高 2 位定义； Bit2-bit7---预留 串行时钟频率计算公式中的 GSR 由 I2C _TGSR_2 中定义。

8.2.3.5 状态映射寄存器（SSR）

表格 8-6 状态映射寄存器功能

状态映射寄存器			功能描述
I2C_SSR_1	Bit0	SRW	位定义参考状态寄存器(SR)中 I2C_SR_1。
	Bit1	SACK	
	Bit2	SI2CB	
	Bit3	SBB	
	Bit4	SDT	
	Bit5	SDR	
	Bit6	SBERR	
	Bit7	SSTOP	
I2C_SSR_2	Bit0	SSAM	位定义参考状态寄存器(SR)中 I2C_SR_2。
	Bit1	SGC	
	Bit2	SAL	
	Bit3	SSTART	
	Bit4~ Bit7		

8.2.4 串行接收、发送缓冲区(DR)

表格 8-7 数据缓冲区功能

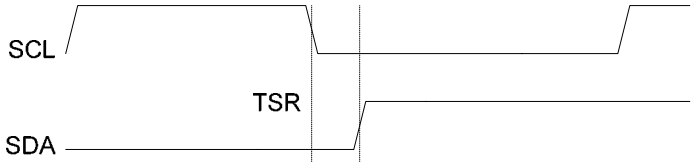
数据缓冲区	功能描述
DR	单字节串行接收、发送数据缓冲区

8.2.5 Setup/Hold 延时及尖脉冲压制 (TGSR)

表格 8-8 TGSR 功能

TGSR			功能描述
I^2C _TGSR_1	Bit0~Bit7	TSR	延时等待位定义 ;TSR 寄存器共 10 位，I^2C _TGSR_1 为 TSR 寄存器的低 8 位；
I^2C _TGSR_2	Bit0~Bit1	TSR	延时等待位定义；I^2C _TGSR_2 中的 BIT0~BIT1 为 TSR 寄存器的高 2 位；
	Bit2~Bit4	GSR	用于计算 SCL 时钟频率信号；
	Bit5~ Bit7		预留；

TSR 的延时等待时间计算举例说明：
40MHz 工作频率（25ns），根据 TSR 的值设置（M）* 25 即得到延时等待时间



8.3 SPI 通讯口

8.3.1 SPI 通讯口描述

SPI（Serial Peripheral Interface）接口为通用的同步串行通讯接口，此接口为 RISE3301 内核与外围带有 SPI 通讯接口固件的通讯桥梁；可根据用户软件设置为主控模式和从动模式，处于主控模式时可外挂 4 个从动 MCU 设备。

SPI 模式同步发送和接收单字节数据，硬件连接如下：

串行数据输出 MOSI-----GPIO_5

串行数据输入 MISO----- GPIO_6

串行时钟 SCLK------ GPIO_4

从动片选: SS_0(GPIO_7)、SS_1(GPIO_8)、SS_2(GPIO_9)、SS_3(GPIO_10)

8.3.2 SPI 口原理框图及工作模式

串行外围接口所要求的四种信号分别为 SCLK、MOSI、MSIO、SS；当 SPI 处于主控模式时，MOSI（master output/slave input）I/O 做为数据输出口使用，当 SPI 处于从动模式时则作为数据输入口使用；在硬件连接上，MOSI 口与从设备的 MOSI 口连接；相反的，当 SPI 处于主控模式时，MISO（master input/slave output）I/O 则作为数据输入口，当 SPI 处于从动模式则作为数据输出口使用；在硬件连接上，MISO 口与从设备的 MISO 口连接；从动片选输出高电平有效。具体设计时若外围器件为低电平有效则需加个反相器。

SPI 端口主要完成 RISE3301 内 SPI 硬件与外围硬件 SPI 的通讯控制功能，SPI 端口由下列模块组成： TX buffer 、RX buffer、 Port Control(PC)register、 baud rate(BR)register 、status(ST)register、 transfer control(TC)register。

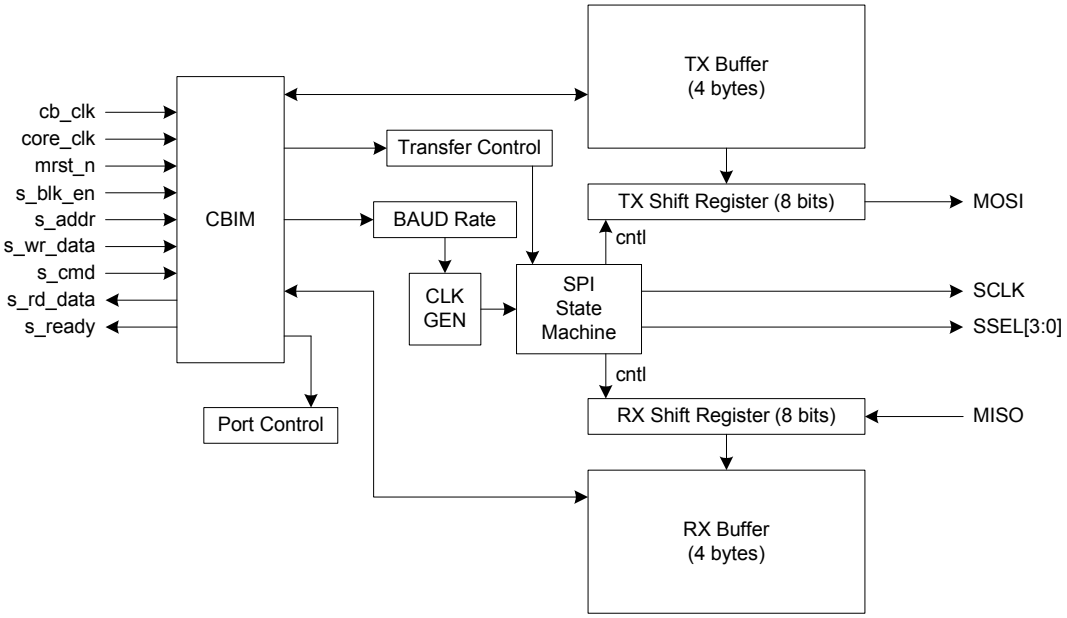


图 8-2 SPI 端口图

SPI 模块各有 4Byte 发送和接收缓冲区，处理器通过设置 Byte_Count 寄存器值来确定发送的字节数；Tx buffer 被赋值后，处理器置发送控制寄存器（TC0）中的发送位，TX buffer 中数据自动被移到 Tx shift register 寄存器中发送，采用单字节（8 位）发送，先发送字节高位，后发送低位；MISO 接收位入 Rx shift register 寄存器中，单字节接收完整后移入 RX Buffer 中。

SPI 模块在发送和接收数据的模式下产生两个中断，当发送数据的最后一个字节移入 Tx shift buffer 寄存器中后 SPI 模块会产生中断，处理器通过发送中断判断数据发送是否完毕并准备下组数据发送的缓冲；另一个中断在最后一个 Rx shift register 寄存器数据移入接收数据缓冲后产生，处理器通过接收中断判断数据是否接收完毕并准备读取缓冲中数据。两个中断使能位在 Port Control(TC)register 中，由软件定义打开和关闭，复位状态时该中断使能位被屏蔽，需要软件置位使能。

外围从设备通过硬件接口与 RISE3301 的 SS[3：0]相连接；处理器通过设置发送寄存器中的 SSEL[3：0]的位定义来确定处理器下一步要与哪个从动 MCU 产生数据通讯。具体寄存器定义说明见下面的工作寄存器定义。

发送字节数寄存器 byte_count[1:0]由软件定义；当发送模块准备发送数据时先启动 Transfer Control 寄存器中的起始位初始化发送模块，初始化结束后 Transfer Control 寄存器中的起始位被 SPI State Machine 自动清零； Port Control(PC)中的连续收发使能位被置位后，SPI State Machine 将会在连续收发模式下工作，即在一组数据发送完毕后立即启动另一组数据的起始发送时序，而不需要软件重新配置 Transfer Control 寄存器中的起始位；装载 Tx buffer 数据缓冲时不需要空指令循环做延时等待，对于工作在连续接收模式下在 RX 中断产生后可立即读取接收缓冲区数据。连续收发模式受控制于 Transfer Control 寄存器中的 STOP 停止位和 Port Control(PC)中的连续收发使能位；两者之一被使能则中止连续工作模式，收发时序在收发完当前数据后则停止工作。

SPI 时钟的级性和与数据发送同步的相位关系分别由有处理器控制 PC 寄存器的 clk_polarity 和 clk_phase 所控制。

SPI 主控模式下的时钟信号工作频率由 SPI_BAUD[7:0]寄存器值换算得到，具体换算公式如下：

以 40M 的 clk_run 为说明：SCLK frequency =5MHz ÷(SPI_BAUD[7:0] +1) ；

得到的 SCLK 时钟频率为 5MHz~19.531KHz

8.3.3 SPI 寄存器

表格 8-9 SPI 寄存器

SPI 寄存器描述(Base Addr: 0x007000)				
偏移地址 (HEX)	寄存器名称	寄存器符号	类型	Reset
0000	SPI 控制寄存器	SPI_CR	R/W	0x00
0001	SPI 波特率寄存器	SPI_BR	R/W	0x00
0002	SPI 收/发控制寄存器	SPI_TCR	R/W	0x00
0003	SPI 状态寄存器	SPI_SR	R	0x00
0004	SPI 数据发送寄存器 1	SPI_TBR_1	R/W	0x00
0005	SPI 数据发送寄存器 2	SPI_TBR_2	R/W	0x00
0006	SPI 数据发送寄存器 3	SPI_TBR_3	R/W	0x00
0007	SPI 数据发送寄存器 4	SPI_TBR_4	R/W	0x00
0008	SPI 数据接收寄存器 1	SPI_RBR_1	R	0x00
0009	SPI 数据接收寄存器 2	SPI_RBR_2	R	0x00
000a	SPI 数据接收寄存器 3	SPI_RBR_3	R	0x00
000b	SPI 数据接收寄存器 4	SPI_RBR_4	R	0x00

8.3.3.1 SPI 控制寄存器

表格 8-10 SPI 控制寄存器功能

控制寄存器			功能描述
SPI_CR	Bit0	RX_INT_EN	数据发送中断使能控制位； “1” = 打开 “0” = 关闭
	Bit 1	TX_INT_EN	数据接收中断使能控制位； “1” = 打开 “0” = 关闭
	Bit 2	CNT_MODE	连续工作模式控制使能位； “1” = 打开 “0” = 关闭
	Bit 3	CLK_PHASE	该标志位控制SCLK信号和MOSI输出数据的相位关系； “0” = 0 相位 “1” = 180 度相位偏移
	Bit 4	CLK_POLAR	该标志位控制输出时钟信号SCLK极性； “0” = 正常模式 “1” = 翻转模式
	Bit 5	RX_INT_CLR	清数据接收状态位，只能写； “1” = 清状态位 “0” = 不起作用
	Bit 6	TX_INT_CLR	清数据接收状态位，只能写； “1” = 清状态位 “0” = 不起作用
	Bit 7		预留；

8.3.3.2 波特率寄存器

用于波特率发生器换算波特率。

表格 8-11 波特率寄存器功能

波特率寄存器	功能描述
SPI_BAUD	波特率寄存器。SCLK Frequency = 20MHz / (SPI_BAUD[7:0] +1)；

8.3.3.3 SPI 收/发控制寄存器

表格 8-12 SPI 收/发控制寄存器功能

收/发控制寄存器			功能描述
SPI_TCR	Bit0	STOP	停止控制位； 置 1 表示 SPI 停止发送数据； 当 SPI 处于连续收发模式，该标志位不起作用；
	Bit1	START	启动控制位； 置 1 表示 SPI 开始发送数据； 该标志位在启动发送后自动清零；
	Bit2~Bit3	BYTE_COUNT	收/发数据字节数；起始位之后的数据；
	Bit4~Bit7	DEV_SELECT	从设备选择； DEV_SELECT[4] = SS[0]； DEV_SELECT[5] = SS[1]； DEV_SELECT[6] = SS[2]； DEV_SELECT[7] = SS[3]；

8.3.3.4 状态寄存器

表格 8-13 状态寄存器功能

状态寄存器			功能描述
SPI_SR（只读）	Bit0	TX_DONE	发送标志位； SPI 最后一个字节被移入 TSR（Transmit Shift Register）后该标志位被置 1；下一次新的发送数据包的第一个字节被移入 TSR 后该标志位被清零；
	Bit1	RX_DONE	接收标志位； SPI 最后一个字节从 RSR（receiver Shift Register）移入 RBR 后该标志位将被置 1；该标志位的清零方式为：当 START 标志被置位，在连续接收模式下每个发送数据包的第一个字节将被移入 RBR；
	Bit2	BUSY	工作状态标志位；
	Bit3	Tx_irq	发送中断标志位； 1---中断产生 0---无中断

	Bit4	Rx_irq	接收中断标志位； 1---中断产生 0---无中断
	Bit5~ Bit7		预留；

8.3.4 发送数据缓冲区

表格 8-14 发送数据缓冲区功能

发送数据缓冲区	功能描述
SPR_TBR_1	四字节发送缓冲数据，紧跟“ START“ 发送数据的次序为 SPR_TBR_1~ SPR_TBR_4，具体发送字节数由 BYTE_COUNT 软件编程定义。 当软件定义发送字节数时，BYTE_COUNT 分别为 1、2、3、4 的发送次序如下： BYTE_COUNT = 1 ;“ SPR_TBR_1 ” BYTE_COUNT = 2 ;“ SPR_TBR_2、SPR_TBR_1 ” BYTE_COUNT = 3 ;“ SPR_TBR_3、SPR_TBR_2、SPR_TBR_1 ” BYTE_COUNT = 4 ;“ SPR_TBR_4、SPR_TBR_3、SPR_TBR_2、SPR_TBR_1 ”
SPR_TBR_2	
SPR_TBR_3	
SPR_TBR_4	

8.3.5 接收数据缓冲区

表格 8-15 接收数据缓冲区功能

接收数据缓冲区	功能描述
SPR_RBR_1	四字节接收缓冲数据，紧跟“ START ”接收数据的次序为 SPR_RBR_1~ SPR_RBR_4，具体接收字节数由 BYTE_COUNT 软件编程定义。
SPR_RBR_2	
SPR_RBR_3	
SPR_RBR_4	

8.4 Timer 定时器

RISE3301 两个内核各带有 3 个 16 位 Timer/Counter；其中 APP 端的 T0 定时器已被 APP-API 接口函数使用，如果用户要使用 APP-T0 定时器，则只能与 API 接口函数共同使用，API 接口函数开放 T0 定时器做为 1ms 定时时基来使用。具体使用操作见 API 接口函数说明。

定时器具体工作模式请参见 FR8052。

第九章 内部 RTC

9.1 硬件原理

实时时钟（RTC）可提供独立的秒、分钟、小时以及天数计数器功能。每一个计数器在其特定的时间期限内触发一次。例如，秒计数器每一秒钟触发一次，分钟计数器每一分钟触发一次，以此类推。在RTC的帮助下，软件本身并不需要计算分钟、小时和天数信息，它只需读取计数器所取得的值并计算出当前时间。

RTC还可以秒、分钟、小时为单位提供可编程的自动报警功能。当秒时基自动报警功能被启动后，RTC将每秒钟自动触发一次中断。同理，当启动分钟、小时时基自动报警功能后，RTC也将在相应的时间触发中断。

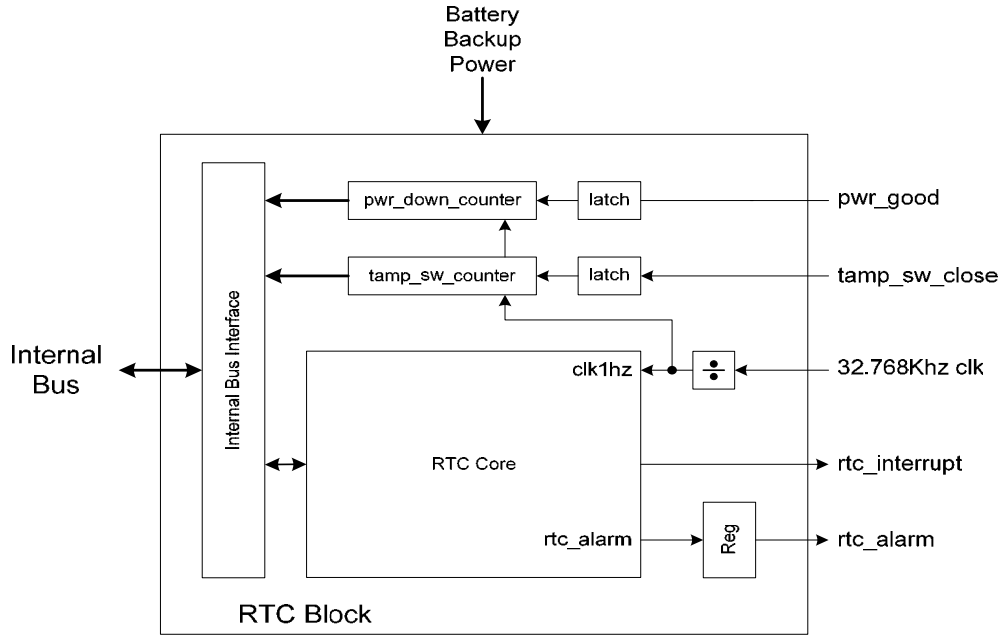


图 9-1 内部 RTC 原理图

9.2 功能描述

9.2.1 实时计时功能

RISE3301 提供片内 RTC 功能，RTC 模块工作电平 Battery Backup Power (2.5VBK 为不间断性工作电源)，由外部提供的 RTC 时钟晶振（32.768KHz）。在停电状态下，若要给 2.5VBK 处供电，则需加时钟电池 BAT 以保证 RTC 运行 。时钟计数器包括 Sec（RTC_SEC）、Min(RTC_MIN)、Hour(RTC_HR)单字节寄存器，Data(RTC_DAY_1、RTC_DAY_2)双字节寄存器；其中 Sec、Min 寄存器时间累计范围在 0-59，Hour 寄存器累计范围为 0-23，Data 寄存器的累计时间可达到 179.5 年，即 65536 天（0xffff）。RTC-Core 硬件数字逻辑电路都有计时寄存器超时自动翻转进位功能；该 RTC 时钟模块具有中断（rtc_interrupt）和比较报警（Ararm）输出功能；具体实现方法由寄存器配置完成。内部所有寄存器计时功能都以秒时基为计时单位。

9.2.2 定时累计功能

在停电时 RTC 的硬件模块 Power Down Counter (pwr_down_counter)， 和 Tamper Switch Counter (tamp_sw_counter)的工作电平由 2.5V BACKUP POWER 管脚提供，当管脚 POWER_GOOD、TMPR_SW_CLOSED 管脚电平从高-低发生变化时，则分别启动 pwr_down_counter、tamp_sw_counter 寄存器以开始计时，激活 RTC 计时累计功能；pwr_down_counter 寄存器将累计单次停电时间，最长的累计时间为 194 天（共三字节 0xffffff）；tamp_sw_counter 寄存器将会累计外部事件发生到结束的总时间，最长累计时间为 136 年（共四字节 0xffffffff）。

9.2.3 内部 RTC 应用说明

1) 硬件连接说明：
   1、2.5 V BK---该电源为不间断性电源，提供 RTC 计时模块电源；
   2、PWR_GOOD，TAMP_SW_CLOSE 引脚电平发生变化时则启动计时功能；
   3、Rtc_alarm 引脚输出信号由软件设置为用户所需求的信息；
   4、Rtc_32KHz 引脚输出供用户使用；
   5、在不使用内部 RTC 的情况下，外挂晶振可悬空；
2) 软件初始化说明
   RTC 计时初始化步骤：
   1、首先对 RTC 寄存器初始化；根据需要设置中断控制及报警比较输出寄存器、RTC_WR 寄存器；
   2、置寄存器 RTC_INTERNAL_CNTL 中的位 RTC_CNT_LD 为 1，则 RTC 模块会自动把 RTC_WR 里的初始值导入 RTC 计时寄存器中；
   3、置寄存器 RTC_INTERNAL_CNTL 中的位 RTC_EN 为 1；则启动 RTC 计时寄存器时会从当前值开始计时；需要注意的是该 RTC 不带年份，需要软件进行计算。

9.3 工作寄存器

控制寄存器的逻辑地址如下表：

表格 9-1 RTC 寄存器

RTC 寄存器：（BASE_ADDR:0X6000）				
偏移地址（HEX）	寄存器名称	寄存器符号	类型	Reset
0000	RTC Second Register	RTC_SEC	R	0x00
0004	RTC Minute Register	RTC_MIN	R	0x00
0008	RTC Hour Register	RTC_HR	R	0x00
000c	RTC Day Register_1	RTC_DAY_1	R	0x00
000d	RTC Day Register_2	RTC_DAY_2	R	0x00
0010	RTC Second alarm Register	RTC_SEC_ALRM	R/W	0x3f
0014	RTC Minute alarm Register	RTC_MIN_ALRM	R/W	0x3f
0018	RTC Hour alarm Register	RTC_HR_ALRM	R/W	0x1f

001c	RTC Record Register_1	RTC_RECORD_1	R/W	0x00
001d	RTC Record Register_2	RTC_RECORD_2	R/W	0x00
001e	RTC Record Register_3	RTC_RECORD_3	R/W	0x00
001f	RTC Record Register_4	RTC_RECORD_4	R/W	0x00
0020	RTC Internal Control Register	RTC_INTERNAL_CNTL	R/W	0x00
0024	RTC Write Second Register	RTC_WR_SECOND	R/W	0x00
0028	RTC Write Minute Register	RTC_WR_MINUTE	R/W	0x00
002C	RTC Write Hour Register	RTC_WR_HOUR	R/W	0x00
0030	RTC Write Day Register_1	RTC_WR_DAY_1	R/W	0x00
0031	RTC Write Day Register_2	RTC_WR_DAY_2	R/W	0x00
0034	RTC Interrupt Status Register	RTC_INT_STAT	R/W	0x00
0040	RTC Power Fail Time Counter_1	PWR_COUNTER_1	R	0x00
0041	RTC Power Fail Time Counter_2	PWR_COUNTER_2	R	0x00
0042	RTC Power Fail Time Counter_3	PWR_COUNTER_3	R	0x00
0044	RTC Tamper Switch Open Time Counter_1	SW_COUNTER_1	R	0x00
0045	RTC Tamper Switch Open Time Counter_2	SW_COUNTER_2	R	0x00
0046	RTC Tamper Switch Open Time Counter_3	SW_COUNTER_3	R	0x00
0047	RTC Tamper Switch Open Time Counter_4	SW_COUNTER_4	R	0x00
0048	RTC Power Fail Time Counter Clear	PWR_COUNTER_CLR	W	---
0049	RTC Tamper Switch Counter Clear	SW_COUNTER_CLR	W	---
004A	RTC OSC Switch On/Off	RTC OSC SW	R/W	---

9.3.1 计时寄存器

RTC 计时寄存器，分秒、分、时、日寄存器，日计时寄存器采用双字节累计方式；这类寄存器只支持读取。具体定义见表格9-2。

表格 9-2 计时寄存器功能

计时寄存器	功能描述
RTC_SEC	秒寄存器，位定义为 bit0~bit5，bit6、bit7 预留；计时范围 0~0x3b(0-59)，自动翻转；
RTC_MIN	分寄存器，位定义为 bit0~bit5，bit6、bit7 预留；计时范围 0~0x3b(0-59)，自动翻转；
RTC_HR	时寄存器，位定义为 bit0~bit4，bit5~bit7 预留；计时范围 0~0x17(0-23)，自动翻转；
RTC_DAY_1	日寄存器，双字节，低 8 位；
RTC_DAY_2	日寄存器，双字节，高 8 位；计时范围 0~0xffff，自动翻转；

9.3.2 报警寄存器

报警比较寄存器，根据用户设置初始值，计时寄存器与报警比较寄存器相比较，两者相等则产生报警信号输出。具体定义见表格9-3。

表格 9-3 报警寄存器功能

报警寄存器	功能描述
RTC_SEC_ALRM	秒报警比较值，位定义为 bit0~bit5，bit6、bit7 预留； 例，假如该值被置为 10，则在每分的第 10 秒产生报警信号；如果该设置值超过 59 则不会产生报警信号；
RTC_MIN_ALRM	分报警比较值，同上；
RTC_HR_ALRM	时报警比较值，位定义为 bit0~bit4，bit5~bit7 预留；

9.3.3 时间调整寄存器

该寄存器存储用户设置的当前时间与RTC的计时时间的差值，该值用于校准时间。具体定义见表格9-4。

表格 9-4 Base time 寄存器功能

Base time 寄存器	功能描述
RTC_RE_1	共四字节，从低到高字节排序；用于校准时间用； Bit0~Bit7；
RTC_RE_2	Bit8~Bit15；
RTC_RE_3	Bit16~Bit23；
RTC_RE_4	Bit24~Bit31；

RtcDays*86400 + RtcHour*3600 + RtcMinute*60 + RtcSecond + RtcRecord
= seconds of (Current time - Base time)

Base time 对应的是初始值寄存器中的值，该值由软件设置；

RtcDays、RtcHour、RtcMinute、RtcSecond、RtcRecord 寄存器是 RTC 从复位状态未经过赋初始值的计时时间；

Current time 用户设置的当前时间。

seconds of (Current time - Base time) 该值是当前时间与 RTC 计时时间的一个差值，存在时间调整寄存器中。

9.3.4 RTC 中断控制寄存器

RTC中断控制寄存器位定义控制RTC的使能和分时报警使能位，具体见表格9-5。

表格 9-5 RTC 控制寄存器功能

RTC 控制寄存器			功能描述
RTC_INT_CNTL	Bit0	RTC_EN	RTC 中断使能控制位； 0：关掉 1：使能
	Bit1	RTC_SEC_INT	秒时基中断报警输出控制位； 0：关掉 1：使能
	Bit2	RTC_MIN_INT	分钟计时中断报警输出控制位； 0：关掉 1：使能
	Bit3	RTC_HR_INT	小时计时中断报警输出控制位； 0：关掉 1：使能
	Bit4	RTC_DAY_INT	日计时中断报警输出控制位； 0：关掉 1：使能
	Bit5	RTC_ALRM_INT	RTC 中断报警输出控制位； 0：关掉 1：使能
	Bit6	RTC_CNT_LD	启动 RTC 时把存放在 RTC_WR_TIME

			寄存器中的初始值附给 RTC 计时寄存器； 0：屏蔽 1：允许
	Bit7	Bit7	预留；

9.3.5 初始值寄存器

该寄存器用于将计时寄存器的值初始化，启动RTC控制寄存器中的RTC_CNT_LD后则把初始值寄存器的值导入计时寄存器中。具体见表格9-6。

表格 9-6 初始值寄存器功能

初始值寄存器	功能描述
RTC_WR_SEC	位定义为 bit0~bit5，bit6、bit7 预留；使能 RTC_CNT_LD 位硬件自动把该寄存器值附给 RTC_REC，使 RTC_REC 有个初始计数值；
RTC_WR_MIN	位定义为 bit0~bit5，bit6、bit7 预留；同上；
RTC_WR_HR	时寄存器，位定义为 bit0~bit4，bit5~bit7 预留；
RTC_WR_DAY_1	时寄存器，位定义为 bit0~bit7；
RTC_WR_DAY_2	时寄存器，位定义为 bit0~bit7；

9.3.6 RTC 状态寄存器

位定义标志定义各种报警状态，该状态寄存器必须由软件清零。具体见表格9-7。

表格 9-7 RTC 状态寄存器功能

RTC 状态寄存器			功能描述
RTC_INT_STAT	Bit0	RTC_SEC_INT	秒中断报警状态；0：无 1：有
	Bit1	RTC_MIN_INT	分中断报警状态；0：无 1：有
	Bit2	RTC_HR_INT	时中断报警状态；0：无 1：有
	Bit3	RTC_DAY_INT	日秒中断报警状态；0：无 1：有
	Bit4	RTC_ALRM	RTC 中断报警状态；0：无 1：有
	Bit5	Bit5	预留；
	Bit6	Bit6	
	Bit7	Bit7	

9.3.7 停电计数寄存器

RTC 模块配有备用电池，可在停电情况下供电，PWR_ COUNTER 停电计数寄存器在引脚 PWR_GOOD(#PIN15)失电后开始计时以秒时基为计时单位；计时范围为（0xffffff）；具体见表格 9-8 定义；在 PWR_GOOD（#PIN15）得电后，通过对寄存器 PWR_COUNTER_CLR(0x6048)进行写操作就可实现 PWR_COUNTER 寄存器清零目的。

表格 9-8 停电计数寄存器功能

停电计数寄存器	功能描述
PWR_ COUNTER_1	共三字节从低到高排序；位定义为 bit0~bit7；停电计数寄存器；
PWR_ COUNTER_2	位定义为 bit8~bit15；
PWR_	位定义为 bit16~bit23；

COUNTER_3	

RTC 模块在停电情况下，可使用备用电池供电。SW_COUNTER 停电计数寄存器在引脚 TMPR_SW_CLOSED(#PIN21)失电后开始计时，以秒时基为计时单位；计时范围为（0xffffffff），具体见表格 9-9 定义；在 TMPR_SW（#PIN21）得电后，通过对寄存器 SW_COUNTER_CLR(0x6049)进行写操作就可实现 SW_COUNTER 寄存器清零目的。

表格 9-9 停电计数寄存器功能

停电计数寄存器	功能描述
SW_COUNTER_1	共四字节从低到高排序；位定义为 bit0~bit7；停电计数寄存器；
SW_COUNTER_2	位定义为 bit8~bit15；
SW_COUNTER_3	位定义为 bit16~bit23；
SW_COUNTER_3	时寄存器，位定义为 bit24~bit31；

9.3.8 开关晶振寄存器

RTC模块在停电情况下，可使用备用电池供电。为了防止设备因久置不用而耗尽电池电量，可以对RTC OSC SW(0x604a)寄存器进行操作。写入“ 0 ”关闭，此时整个RTC处于关闭状态，晶振停止，耗电量极低。写入非 “ 0 ” 的数据则开启RTC，晶振起振。

第十章 复位

RISE3301 芯片复位方式有外围硬件复位，底层软件控制复位、WDT 复位。

10.1 外围硬件复位

外围硬件复位信号(ext_reset_n)由外围器件提供，使系统复位；

外围硬件复位信号的要求为：脉冲的宽度最小为 Ts150ns,最小压差为 Vpp 2.7V.

波形如下：

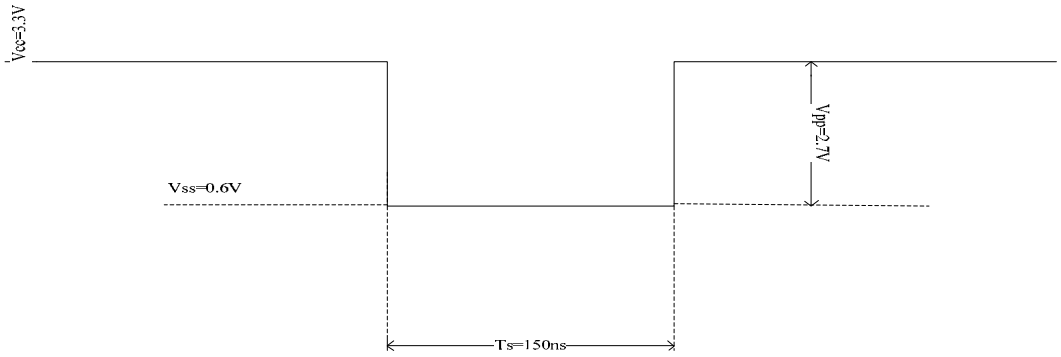


图 10-1 外围硬件复位信号波形

10.2 底层软件控制复位

上图中“ sw_rst_X ”复位信号的产生由软件配置对应的控制寄存器来实现；配置对应的控制寄存器位即可实现相应的外设 (I^2C、SPI、RTC、etc)复位。具体寄存器定义见表格 10.4 章节。

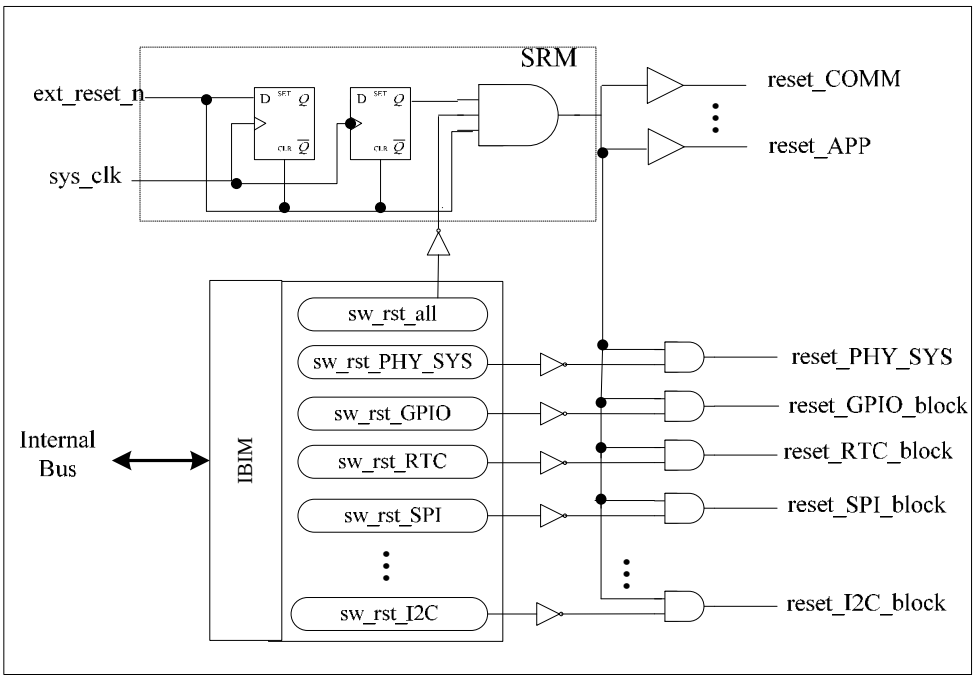


图 10-2 复位图

10.3 WDT 复位

当 watchdog 使能，watchdog 定时时间发生溢出，则产生 MCU 复位；具体看门狗定义可查看用户手册，内部看门狗使用可查看 API 接口函数关于看门狗的调用说明。

10.4 复位寄存器

复位寄存器位定义控制相对应的硬件模块复位。

表格 10-1 复位寄存器位定义

系统 RESET 模块寄存器定义（BASE ADDR：0x009000）				
偏移地址	寄存器名称	寄存器符号	类型	复位状态
0000	系统 RESET 控制寄存器 1	SYS_RESET_CNTL_1	R/W	0xf0
0001	系统 RESET 控制寄存器 2	SYS_RESET_CNTL_2	R/W	0xfd

表格 10-2 系统 RESET 控制寄存器功能

系统 RESET 控制寄存器 1			功能描述
SYS_RESET_CNTL_1	Bit0	SW_RESET_ALL	“1”：复位整个芯片；
	Bit1	PHY_SYS_RST_0	“1”：底层系统复位控制位；
	Bit2	保留	
	Bit3	保留	
	Bit4	PHY_SYS_RST_3	
	Bit5	PHY_SYS_RST_4	
	Bit6	PHY_SYS_RST_5	
	Bit7	I^2C _RESET	“1”：复位 I^2C 模块；

系统 RESET 控制寄存器 1			功能描述
SYS_RESET_CNTL_2	Bit0	RTC_RESET	“1” :复位 RTC 模块硬件控制逻辑；
	Bit1	REC_CORE_RESET	“1”：复位 RTC 核；
	Bit2	SPI_RESET	“1”：复位 SPI 模块；
	Bit3	PHY_SYS_RST_6	“1”：底层系统复位控制位；
	Bit4	PHY_SYS_RST_7	
	Bit5	GPIO_RESET	“1”：复位 GPIO 接口模块；
	Bit6	PHY_SYS_RST_8	“1”：底层系统复位控制位；
	Bit7	PHY_SYS_RST_9	

注： **1.**如用户对“底层系统复位控制位”进行复位操作将会引起底层通信的中断以至整个系统的死机；强烈建议用户只操作 I^2C、RTC、SPI、GPIO 模块复位的控制位，同时在对 I^2C、RTC、SPI、GPIO 模块进行复位操作后，必须使用软件对该控制位清零以完成复

位功能。

2.COMM 底层软件复位后对 SYS_RESET_CNTL_1、SYS_RESET_CNTL_2 的初始化分别为 0x00、0x00;

3.从上面的定义可以看出，有些位定义会引起系统模块以及 COMM 底层通讯的复位，所以用户在操作此寄存器时要谨慎，应采用位操作的方式来实现。

第十一章 电气特性

11.1 工作参数

表格 11-1 工作参数表

符号	参数	最小值	典型值	最大值	参数
V_{CK}	数字核工作电压 2.5V	2.25	2.50	2.75	V
V_{IO}	I/O 口电压 3.3V	3.0	3.3	3.6	V
V_{CC3A}	模拟工作电压 3.3V	3.0	3.3	3.6	V
Tj	工作温度范围	-40		85	℃

11.2 极限参数

表格 11-2 极限参数表

符号	参数	范围	单位
V_{CK}	Core Power Supply	-0.3 ~ +3.05	V
V_{IO}	3.3V I/O Input Voltage	-0.3 ~ +3.9	V
Tjmax	Maximum Junction temperature	+125	℃
Tstg	储存温度	-40 到+150	℃

11.3DC 特性

表格 11-3 DC 特性表

符号	参数	条件	最小	类型	最大	单位
I_{IN}	Input current	No pull-up or pull-down	-10	± 1	10	uA
Ioz	Tri-state Leakage current		-10	± 1	10	uA
C_{IN}	Input Capacitance			3.1		pF
C_{OUT}	Output Capacitance			3.1		pF
V_{IL}	2.5V Input low voltage	CMOS			0.7	V
	3.3V Input low voltage				0.8	
V_{IH}	2.5V Input high voltage	CMOS	1.7			V
	3.3V Input high voltage		2.0			
V_{OL}	2.5V Output low voltage	$\|I_{OL}\|$ = 2 ~ 16mA			0.4	V
	3.3V Output low voltage				0.4	
V_{OH}	2.5V Output high voltage	$\|I_{OH}\|$ = 2 ~ 16mA	1.85			V
	3.3V Output high voltage		2.4			
DAC_XV	Tx DAC Top reference				2.7	V

RT	voltage					
DAC_XVRB	Tx DAC Bottom reference voltage		0.3			V
DAC_VOUT	Tx DAC Output Voltage		0.33		2.7	V
DAC_Load	Tx DAC Capacitive load				15	pF
PGC_VCM	Programmable Gain Control Input Common Mode Voltage			1.65		V
PGC_Rin	Programmable Gain Control Input impedance		14			KΩ
PGC_Vin	Programmable Gain Control Input voltage range		0.33		2.97	V
ADC_REFN	Rx ADC Bottom reference voltage			0.75		V
ADC_REFP	Rx ADC Top reference voltage			1.75		V
R_{UP}	Input Pull-up Resistance		40	75	190	KΩ
R_{DN}	Input Pull-down Resistance		40	75	190	KΩ
Pwr	Total Power Consumption	f_{osc} = 10MHz，pll-8；		70		mA

11.4 时序特性

标定工作参数下的时序特性为：

V_{CK} = +2.25V to +2.75V、 V_{IO} = +3.0V to +3.6V、 f_{CB_CLK} = 80MHz (50% 占空比)；C_{load}=50pF、T_A =-40~+85 °C、在 T_A = +25°C、 V_{CK} = +2.5V、V_{IO} = +3.3V 条件下的典型值。

11.4.1 Clocks Timing（时钟定时）

表格 11-4 时钟定时

符号	参数	最小	类型	最大	单位
OSC_10M_IN	10MHz Oscillator Clock input	-20PPM	10.0	+20PPM	MHz
OSC_32K_IN	32KHz Oscillator Clock input	-5PPM	32.0	+5PPM	KHz
SCL	I^2C Bus Clock Rate	16.67		400	KHz
SCLK	SPI Bus Clock Rate	19.5		5000	KHz

11.4.2 I²C Interface Timing（I²C 接口定时）

表格 11-5 I²C 接口定时

符号	参数	最小	类型	最大	单位
T1	SCL clock frequency	0		400	KHz
T2	Hold time START condition. After this period， the first clock pulse is generated	0.6			μs
T3	Low period of the SCL clock	1.3			μs
T4	High period of the SCL clock	0.6			μs
T5	Set up time for a repeated START condition	0.6			μs
T6	Data hold time	0		0.9	μs
T7	Data set-up time	100			ns
T8	Set-up time for STOP condition	0.6			μs

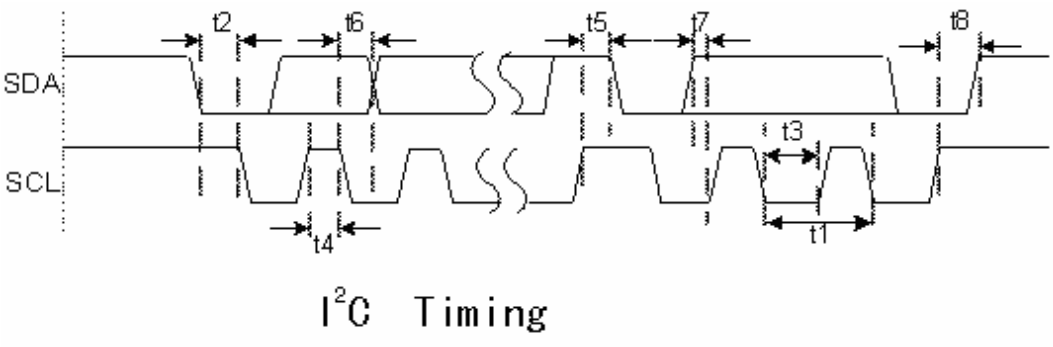


图 11-1 I²C 定时

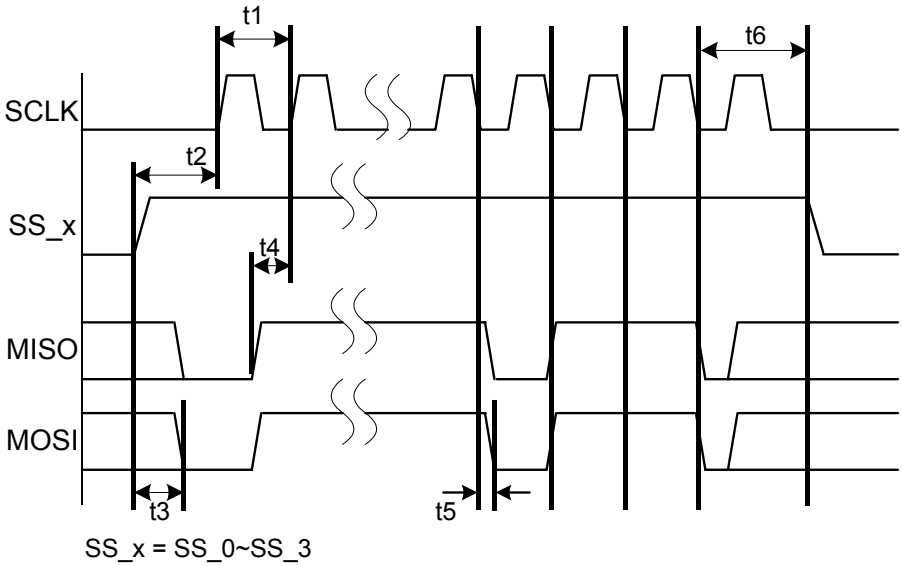
11.4.3 SPI Interface Timing （SPI 接口定时）

表格 11-6 SPI 接口定时

符号	参数	最小	类型	最大	单位
T1	SCLK clock time period^{-1}	0.2		51.2	us
T2	SS_x active to the first rising edge of SCLK clock			1	T1^{-2}
T3	SS_x active to MOSI output valid			0.5	T1
T4	MISO input setup time before rising edge of SCLK	25			ns
T5	MOSI output delay time after falling edge of SCLK			10	ns
T6	Last MOSI output data valid to SS_x de-inserted.			1	T1

注：1.SCLK 时钟波特率可通过 SPI_BAUD 寄存器进行编程。有关如何配置 SCLK 波特率的详细信息，请参阅本文档的 SPI 部分。

2.上表中作为单位的 T1 是指由 SCLK 时钟期的数量所测得的时间。



SPI Timing

图 11-2 SPI 定时

注：在读取数据时，由于发送完一个读取命令后 SCLK 会立即变为低电平，不能够继续提供 CLK 信号，因此，为了给线路继续提供 CLK 信号，需要在命令发送完毕时继续不间断地发送数据(数据可以随意取值)，见下例。

范例：

```
MOV     A,#05H
MOV     DPTR,#SPI_TBR_1
MOVX    @DPTR,A             //数据发送寄存器 1 的内容：05H
INC     DPTR
MOVX    @DPTR,A             //数据发送寄存器 2 的内容：05H
MOV     DPTR,#SPI_TCR
MOVX    A,@DPTR
ANL  A,#0F3H
ORL     A,#04H
MOVX    @DPTR,A     //SPI_SR:DEV_SECLECT=SS[0],BYTE_COUNT=2，选择
                    设备 0，//缓冲区设置：2 个字节
MOV     DPTR,#SPI_TCR
MOVX    A,@DPTR
```

```
    ORL     A,#02H      //SPI_TCR:xxxx xx10  ,启动发送
    MOVX    @DPTR,A
TX_OK_CHK1:
    MOV     DPTR,#SPI_SR
    MOVX    A,@DPTR
    ANLA,#04H
    JNZ     TX_OK_CHK1
    MOV     DPTR,#SPI_CR            ///清中断标志
    MOVX    A,@DPTR
    ANLA,#60H
    MOVX    @DPTR,A
```

本例中第一个 05H 为读取命令，随后发送的 05H 只为线路提供一个 Byte 的 CLK 信号，这个数据不用接收。如果需要读取多个数据，则需要在发送读取命令后再发送相应个数据，为线路提供 CLK 信号。

免责声明

瑞斯康微电子保留其对芯片产品的参数和功能进行修改及更正的权利。瑞斯康微电子建议用户在定货前联系瑞斯康微电子有关部门以获得最新版本的芯片数据手册和相应的参考应用电路。

瑞斯康微电子声明已尽最大可能通过多种仿真以及实验测试手段对芯片以及相关软件和外围参考应用电路的功能和性能进行了尽职地检查测试以确保其芯片性能达到本数据手册所记录的参数。用户是决定在其产品或系统中是否采用瑞斯康微电子芯片的最终决策者，并对其最终产品或系统以及通过使用本芯片产品所带来的结果负责，为了减少风险，瑞斯康微电子慎重建议用户在将瑞斯康微电子芯片用于某特定的实际应用之前，进行充分的实地试验以确保芯片能满足该实际应用的所有要求。

在使用本芯片产品期间，由于本公司合理控制范围以外的原因，包括但不限于自然灾害、罢工或骚乱、暴动、战争行为、政府行为、通讯或其他设施故障或严重伤亡事故等，所造成的财产损失或一切后果，瑞斯康微电子不对您承担任何责任。

未经瑞斯康微电子以及当地有关机构的书面许可，不能将瑞斯康微电子芯片用于可能会涉及人员伤亡以及造成重大财产损失和环境污染的产品或系统，如生命支持系统等；因此，瑞斯康微电子不对因擅自使用本芯片产品所造成的一切不良后果负责，由相关用户承担由此引发的全部法律和经济责任。

瑞斯康微电子（深圳）有限公司保留所有权利。

附录

版本修订记录

名称	Rev.	发行日期	修订内容			修订日期
			章节	页	修订要点	
RISE3301 数据手册	1.1					2008-4-10

瑞斯康微电子(深圳)有限公司

RISECOMM MICROELECTRONICS (SHENZHEN) CO.,LTD

地址：深圳市南山区高新技术园南区创维大厦 C 座 501

邮编：518057

电话：+86-755-33955361 33955359 33955353

传真：+86-755-33955350

邮箱: marketing@risecomm.com.cn